Contents

Getting Started .......................................................... 6
   Supported Platforms ................................................. 6
   End of Support Announcements for Crucible ....................... 7
   End of Support Announcement for IBM ClearCase ................. 10
Installing Crucible on Windows ...................................... 10
   Running Crucible as a Windows service ......................... 13
Installing Crucible on Linux and Mac ............................... 17
Configuring JIRA Integration in the Setup Wizard ................. 20
Starting to use Crucible ............................................ 28
Using Crucible ................................................................ 35
   The Crucible workflow .............................................. 36
Using the Crucible Screens ........................................... 43
   Browsing All Reviews .............................................. 45
   Browsing Source Files ............................................. 46
   Using the Dashboard ............................................... 48
Crucible Icons ................................................................ 53
Searching Crucible ...................................................... 54
Viewing People’s Statistics in Crucible .............................. 59
Browsing Projects ......................................................... 62
   Viewing Project Statistics ......................................... 64
Viewing Reports .......................................................... 66
   Review Coverage Report ......................................... 69
   Changeset Discussions ............................................. 73
Changing your User Profile ........................................... 75
Roles and Status Classifications ..................................... 75
Conducting a Review ..................................................... 76
   Creating a Review ................................................... 77
      Creating a Patch Review ........................................ 77
      Creating a Review within Crucible ............................ 85
      Creating a review from FishEye ................................. 88
      Creating a review from JIRA ................................... 89
      Creating a Review from a URL ................................. 90
      Creating a Snippet Review ..................................... 91
   Selecting the Files for the Review ................................. 92
      Iterative Reviews ................................................ 99
   Adding Reviewers .................................................... 99
Issuing a Review ........................................................ 102
   Performing the Review ............................................. 104
      Adding Comments ............................................... 105
      Flagging Defects ................................................ 107
      Completing your Review ....................................... 108
   Sending a Review’s Comments via Email ....................... 109
   Using JIRA Integration in Crucible Reviews .................... 111
   Using the Review History Dialog ................................. 114
   Tracking Crucible Review Metrics .............................. 115
      Using Progress Tracking ...................................... 115
      Using Time Tracking ............................................ 117
   Summarising and Closing the Review ............................. 118
    Moving a Review to Another Project ......................... 120
    Deleting a review ................................................. 121
   Defining your Workflow ......................................... 122
Viewing Your Favourites .............................................. 124
   Using Favourites .................................................. 126
Using Keyboard Shortcuts in Crucible .............................. 129
Using RSS Feeds in Crucible ......................................... 131
Using Wiki Markup in Crucible ..................................... 131
Using Gadgets in Crucible .................................................... 136
Using Review Reminders ......................................................... 141
Transitioning JIRA issues ......................................................... 141
Administering Crucible ............................................................ 142
Crucible and FishEye ............................................................... 143
Creating a User ......................................................................... 144
Administering projects .............................................................. 145
Creating a Project .................................................................. 146
Setting Crucible to Store all Revisions ...................................... 147
Deleting a Project .................................................................. 148
Editing a Project ................................................................... 148
Enabling the Moderator Role .................................................... 150
Setting Default Review Objectives .......................................... 152
Setting the Default Review Duration for a Project ................... 152
Configuring Repositories .......................................................... 152
Enabling Reviews from the Server File System in Crucible ......... 152
Setting Up a Git Repository in Crucible .................................... 153
Setting Up a Perforce Repository in Crucible Alone ................ 155
Setting Up a Repository via FishEye .......................................... 156
Setting Up a Subversion Repository in Crucible Alone .......... 156
Customising the Welcome Message .......................................... 157
Customising email notifications ................................................. 159
Freemarker Data Model for Email Templates ......................... 160
Setting up Users and Security .................................................. 161
Customising the Defect Classifications .................................... 161
Creating a Permission Scheme ................................................ 164
Agile Permissions Schemes in Crucible .................................... 168
Associating a Permission Scheme with a Project ...................... 168
Configuring User Managed Mappings ...................................... 171
Enabling Access Logging in Crucible ....................................... 171
Backing up and restoring Crucible data .................................... 173
Best Practices for Crucible Configuration ................................ 182
Managing plugins .................................................................. 183
Viewing your Installed Plugins ............................................... 184
Installing a Plugin ................................................................... 185
Configuring a Plugin ............................................................... 187
Uninstalling a Plugin ............................................................... 187
Disabling or Enabling a Plugin ................................................ 187
Checking Plugin Compatibility for Crucible Upgrades ............... 189
Upgrading your Existing Plugins ............................................. 190
Viewing the Plugin Audit Log ................................................... 191
Migrating to an External Database .......................................... 191
Migrating to MySQL Enterprise Server ................................... 192
Migrating to Oracle ................................................................. 195
Migrating to PostgreSQL ......................................................... 198
Migrating to SQL Server .......................................................... 200
Integrating Crucible with Other Applications ......................... 203
Configuring Application Links ............................................... 203
Adding an Application Link ...................................................... 203
Configuring Authentication for an Application Link ................. 206
Configuring Basic HTTP Authentication for an Application Link 209
Configuring OAuth Authentication for an Application Link .... 209
Configuring Trusted Applications Authentication for an Application Link 211
Incoming and Outgoing Authentication .................................... 214
Configuring Project Links across Applications ....................... 214
Adding Project Links between Applications ............................ 215
Deleting a Project Link ............................................................ 217
Making a Project Link the Primary Link .................................... 217
Deleting an Application Link .................................................... 217
Editing an Application Link ...................................................... 218
Making an Application Link the Primary Link ......................... 219
Relocating an Application Link ............................................... 220

Created by Atlassian in 2012. Licensed under a Creative Commons Attribution 2.5 Australia License.
# Getting Started

## Supported Platforms

This page shows the supported platforms for and its minor releases.

**Key:** ✔️ = Supported; ❌ = Not Supported

| **Java Version** |  
|------------------|------------------|
| JRE / JDK (1)    | ✔️ 1.6           |
|                  | ✔️ 1.7           |

<table>
<thead>
<tr>
<th><strong>Operating Systems</strong></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Microsoft Windows (2)</td>
<td>✔️</td>
</tr>
<tr>
<td>Linux (2)</td>
<td>✔️</td>
</tr>
<tr>
<td>Apple Mac OS X (2)</td>
<td>✔️</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Databases</strong></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>MySQL</td>
<td>✔️ MySQL Enterprise Server 5.x</td>
</tr>
<tr>
<td></td>
<td>✔️ MySQL Community Server 5.x</td>
</tr>
<tr>
<td>PostgreSQL</td>
<td>✔️ 8.2, 8.3, 8.4</td>
</tr>
<tr>
<td>Oracle</td>
<td>✔️ 11g</td>
</tr>
<tr>
<td>HSQLDB (3)</td>
<td>✔️ (bundled; for evaluation use only)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Web Browsers</strong></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Microsoft Internet Explorer</td>
<td>✔️ 8.0, 9.0</td>
</tr>
<tr>
<td></td>
<td>❌ 6.0 and 7.0 are not supported</td>
</tr>
<tr>
<td>Mozilla Firefox</td>
<td>✔️ Latest stable version supported</td>
</tr>
<tr>
<td></td>
<td>✔️ 3.6, 4.0</td>
</tr>
<tr>
<td>Safari</td>
<td>✔️ Latest stable version supported</td>
</tr>
<tr>
<td></td>
<td>✔️ 4, 5</td>
</tr>
<tr>
<td>Chrome</td>
<td>✔️ Latest stable version supported</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Version Control Systems</strong></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Subversion (SVN)</td>
<td>✔️ Server 1.1, 1.2, 1.3, 1.4, 1.5, 1.6, 1.7 The client uses SVNkit or native JavaHL.</td>
</tr>
</tbody>
</table>
### Supported Platform Notes

1. Crucible requires **Java Runtime** (JDK or JRE), version as noted in the table above. Pre-release/Early access versions of the Java Runtime are **not supported**.

   You can [download](#) a Java Runtime for Windows/Linux/Solaris. On Mac OS X, the JDK is bundled with the operating system. **Note:** We highly recommend that you use the Oracle JVM (or use the default Mac OS X JVM), as other implementations have not been tested. Please note:
   - Once you have installed the JDK, you need to set the JAVA_HOME environment variable. See [Installing Crucible on Windows](#) or [Installing Crucible on Linux and Mac](#) for details.
   - If you are using a 64-bit JVM, please ensure that you've set your max heap size (`-Xmx`) to a reasonable value, considering the RAM requirements of your system.

2. Crucible is a pure Java application and should run on any platform provided the requirements for the JRE or JDK are satisfied.

3. The Crucible built-in database, running **HSQLDB** is somewhat susceptible to data loss during system crashes. We recommend that you do not use HSQLDB for production systems. External databases are generally more resistant to data loss during a system crash and are more suited for production use.

   FishEye also supports a number of external databases, as noted in the table above. See the [Crucible Database documentation](#) for further details.

### Deployment Notes for Source Code Repositories

Crucible can also store uploaded files in its own database, removing the need for any kind of repository. A number of external databases are supported when Crucible is used with FishEye. See the [FishEye Supported Platforms](#).

### End of Support Announcements for Crucible

This page contains announcements of the end of support for various platforms and browsers when used with Crucible. This is summarised in the table below. Please see the sections following for the full announcements.

#### End of Support Matrix for Crucible

<table>
<thead>
<tr>
<th>Platform</th>
<th>Crucible End of Support</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Created by Atlassian in 2012. Licensed under a Creative Commons Attribution 2.5 Australia License.
The table above summarises information regarding the end of support announcements for upcoming Crucible releases. If a platform (version) has already reached its end of support date, it is not listed in the table.

### Why is Atlassian ending support for these platforms?

Atlassian is committed to delivering improvements and bug fixes as fast as possible. We are also committed to providing world class support for all the platforms our customers run our software on. However, as the complexity of our applications grows, the cost of supporting multiple platforms increases exponentially. Each new feature has to be tested on several combinations of application servers, databases, web browsers, etc, with setup and ongoing maintenance of automated tests. Moving forward, we want to reduce the time spent there to increase Crucible development speed significantly.

### On this page (most recent announcements first):

- [Deprecated Database Support for Crucible (4 October 2011)](#)
- [Deprecated Web Browsers for Crucible (21 March 2011)](#)
- [ Deprecated Java Platforms for Crucible (21 March 2011)](#)
- [Deprecated SCM Repository Support for Crucible (4 April 2011)](#)

### Deprecated Database Support for Crucible (4 October 2011)

This section announces the end of Atlassian support for certain databases for Crucible.

We will stop supporting older versions of databases as follows:

- For the next major version of Crucible, in January 2012, support for MySQL 5.0, PostgreSQL 8.0 and 8.1 will end.

Please refer to the [Supported Platforms](#) for more details regarding platform support for Crucible. If you have questions or concerns regarding these announcements, please email eol-announcement at atlassian dot com.

<table>
<thead>
<tr>
<th>Database</th>
<th>Support End Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>MySQL 5.0</td>
<td>January 2012</td>
</tr>
<tr>
<td>PostgreSQL 8.0 and 8.1</td>
<td>January 2012</td>
</tr>
</tbody>
</table>

### End of Support Notes for MySQL 5.0 and PostgreSQL 8.0 and 8.1:

- Atlassian intends to end of life support for MySQL 5.0, PostgreSQL 8.0 and 8.1 in January 2012. The release of Crucible after January 2012 will not support MySQL 5.0, PostgreSQL 8.0 or 8.1.
- As mentioned above, the releases of Crucible before January 2012 will contain support for MySQL 5.0 and PostgreSQL 8.0 and 8.1.

### Deprecated Web Browsers for Crucible (21 March 2011)
This section announces the end of Atlassian support for certain web browsers for Crucible.

We will stop supporting older versions of web browsers as follows:

- From Crucible 2.6, due in May 2011, support for Internet Explorer 7 will end.

The details are below. Please refer to the Supported Platforms for more details regarding platform support for Crucible. If you have questions or concerns regarding this announcement, please email eol-announcement at atlassian dot com.

### End of Life Announcement for Web Browser Support

<table>
<thead>
<tr>
<th>Web Browsers</th>
<th>Support End Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Internet Explorer 7</td>
<td>When Crucible 2.6 releases (target May 2011)</td>
</tr>
</tbody>
</table>

**Internet Explorer 7 Notes:**

- Crucible 2.5 is the last version to officially support Internet Explorer 7.
- Crucible 2.6 is currently targeted to release in May 2011 and will not be tested with Internet Explorer 7. After the Crucible 2.6 release, Atlassian will not provide fixes in older versions of Crucible for bugs affecting Internet Explorer 7.

### Deprecated Java Platforms for Crucible (21 March 2011)

This section announces the end of Atlassian support for certain Java Platforms for Crucible.

We will stop supporting the following Java Platforms:

- From Crucible 2.6, due in May 2011, support for Java Platform 5 (JDK/JRE 1.5) will end.

We are ending support for Java Platform 5, in line with Sun's Java SE Support Road Map (i.e. "End of Service Life" for Java Platform 5 dated October 30, 2009). We are committed to helping our customers understand this decision and assist them in updating to Java Platform 6, our supported Java Platform.

The details are below. Please refer to the Supported Platforms for more details regarding platform support for Crucible. If you have questions or concerns regarding this announcement, please email eol-announcement at atlassian dot com.

### End of Life Announcement for Java Platform Support

<table>
<thead>
<tr>
<th>Java Platform</th>
<th>Support End Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Java Platform 5 (JDK/JRE 1.5)</td>
<td>When Crucible 2.6 releases, due May 2011</td>
</tr>
</tbody>
</table>

**Java Platform 5 End of Support Notes:**

- Crucible 2.5 is the last version to officially support Java Platform 5 (JDK/JRE 1.5).
- Crucible 2.6 is currently targeted to release in May 2011 and will not be tested with Java Platform 5 (JDK/JRE 1.5). After the Crucible 2.6 release, Atlassian will not provide fixes in older versions of Crucible for bugs affecting Java Platform 5 (JDK/JRE 1.5).

### Deprecated SCM Repository Support for Crucible (4 April 2011)
This section announces the end of Atlassian support for certain SCM repositories for Crucible. End of support means that Atlassian will remove all functionality related to certain SCM repositories past the specified date. Releases before that date will contain the functionality that supports the SCM, however, Atlassian will fix only critical bugs that affect functionality for that SCM, and will not add any new features for that SCM.

Please refer to the Supported Platforms for more details regarding platform support for Crucible. If you have questions or concerns regarding these announcements, please email eol-announcement at atlassian dot com.

<table>
<thead>
<tr>
<th>SCM Repository</th>
<th>Support End Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM ClearCase (all versions)</td>
<td>4 April 2012</td>
</tr>
</tbody>
</table>

IBM ClearCase End of Support Notes:

- Atlassian intends to end of life IBM ClearCase functionality on 4 April 2012. The release of Crucible after 4 April 2012 will not contain any IBM ClearCase functionality.
- As mentioned above, the releases of Crucible before 4 April 2012 will contain support for IBM ClearCase. However, we will only be fixing critical bugs related to IBM ClearCase and will not be adding any feature.

End of Support Announcement for IBM ClearCase

Support in Crucible for IBM ClearCase ended on April 4th 2012. Crucible 2.8, and later versions, do not have support for ClearCase.

We have made these decisions to reduce the testing time required for each release and to help us to deliver market-driven features faster.

You can stay on older versions of Crucible to support your existing installations with ClearCase. However, Atlassian will remove all functionality related to ClearCase, from Crucible versions released after April 4th 2012. We are committed to helping our customers understand this decision and to assist you in migrating to a different SCM, if needed.

For more details about the announcement, please refer to this page: End of Support Announcements for Crucible.

Installing Crucible on Windows

Hey! We're going to perform a clean install of Crucible on Windows. There are a few steps involved, but we think you'll find it easy to follow along.

If you already have FishEye installed, you should read Upgrading from FishEye to Crucible instead.

1. Check supported platforms

Better check the Supported Platforms page first; it lists the application servers, databases, operating systems, web browsers and JDKs that we have tested Crucible with, and that we recommend.

Atlassian only officially supports Crucible running on x86 hardware and 64-bit derivatives of x86 hardware.

Related pages:

- Running Crucible as a Windows service
- Upgrading from FishEye to Crucible
- Installing Crucible on Linux and Mac
- Starting to use Crucible
- Supported Platforms
2. Check your version of Java

In a command prompt, run this:

```
java -version
```

The version of Java should be 1.6.0 or higher. If you intend to Running Crucible as a Windows service, using the Java Service Wrapper, you should use 32-bit Java (even on a 64-bit machine), and the JDK rather than the JRE (so as to take advantage of the `-server` parameter).

**If you don't see Java 1.6.0 or higher, then get Java...**

Download and install the Java Platform JDK from Oracle's website.

⚠️ The Java install path should not contain spaces, so don't install into C:\Program Files\Java. Instead, use a path like C:\Java.

Now try running `java -version` again to check the installation. The version of Java should be 1.6.0 or higher.

3. Check that Windows can find Java

Windows uses the JAVA_HOME environment variable to find Java. To check that, in a new command prompt, run:

```
echo %JAVA_HOME%
```

You should see a path to the Java install location that does not contain spaces. We recommend that JAVA_HOME should point to the Java executable in your PATH.

**If you don't see a path without spaces...**

- If you see a path with spaces, like C:\Program Files\Java, then sorry, but go back to 2. and reinstall Java to a location that doesn't have spaces.
- If you don't see a path at all, or if you just see %JAVA_HOME%, then set JAVA_HOME as follows:

  For Windows 7:

  1. Go to Start, search for "sys env" and choose Edit the system environment variables.
  2. Click Environment Variables, and then New under 'System variables'.
  3. Enter "JAVA_HOME" as the Variable name, and the absolute path to where you installed Java as the Variable value. Don't use a trailing backslash. We recommend that JAVA_HOME should point to the Java executable in your PATH.
  4. Now, in a new command prompt, try running `%JAVA_HOME%\bin\java -version`. You should see the same version of Java as you saw above.

4. Create a dedicated Crucible user (recommended)

For production installations, we recommend that you create a new dedicated Windows user that will run Crucible on your system. This user:

- Should not have admin privileges.
• Should be a non-privileged user with read, write and execute access on the Crucible home (install) directory and instance (data) directory. These directories are described below.
• Should only have read access to your repositories.

If you created a dedicated Crucible user, ensure you are logged in as this user to complete the remaining instructions.

5. Now it’s time to get Crucible

Download Crucible from the Atlassian download site.

Extract the downloaded file to an install location:

• Folder names in the path to your Crucible executable should not have spaces in them. The path to the extracted directory is referred to as the <FishEye home directory> in these instructions, for compatibility with the FishEye install instructions.
• If you expect to have a large number of users for this Crucible installation, and Crucible will be connected to an external database, consider installing Crucible on a different server from the one running the external database, for improved performance.

6. Tell Crucible where to store your data

The Crucible instance directory is where your Crucible data is stored.

⚠️ You should not locate your Crucible instance directory inside the <FishEye home directory> — these should be entirely separate locations. If you do put the instance directory in the <FishEye home directory> it will be overwritten, and lost, when Crucible gets upgraded. And by the way, you'll need separate Crucible instance directories if you want to run multiple copies of Crucible.

Create your Crucible instance directory, and then tell Crucible where you created it by setting a FISHEYE_INST environment variable, as follows:

For Windows 7:

1. Go to Start, search for “sys env” and choose Edit the system environment variables.
2. Click Environment Variables, and then New under ‘System variables’.
3. Enter "FISHEYE_INST" as the Variable name, and the absolute path to your new Crucible instance directory as the Variable value. Don't use a trailing backslash.
4. Now copy the newly extracted <FishEye home directory>\config.xml file to the root of your new FishEye instance directory.

ℹ️ Note that if Crucible is run as a Windows service using the Java Service Wrapper, Crucible-specific environment variables such as FISHEYE_INST are ignored – these must be set in the wrapper.conf file. See Running Crucible as a Windows service.

If you have a large number of repositories, we recommend you increase the default number of files that Crucible is allowed to open. See the following knowledge base article for more info: Subversion Indexer Paused with "Too many open files" Error.

7. Start Crucible!

In a command prompt, change directory to <FishEye home directory> and run this:

```
bin\start.bat
```

After a few moments, in a web browser on the same machine, go to http://localhost:8060/ (or, from another
Enter your license, then an admin password, to finish the setup.

You can postpone setting up JIRA integration until later if you wish; see Configuring JIRA integration in the Setup Wizard.

8. Connect to an external database (recommended)

If you intend to use this Crucible installation in a production environment, it is highly recommended that you use one of the supported external databases. See Migrating to an external database.

If you are evaluating Crucible, or don’t wish to do this now, Crucible will happily use its embedded database, and you can easily migrate later.

9. Set up your mail server

Configure the Crucible email server so that users can get notifications from Crucible. See Configuring SMTP.

10. Add users and repositories

Now is the time to set up your users in Crucible, and to tell Crucible about any existing repositories you have. Please read Starting to use Crucible for the details.

Crucible will perform an initial index of your repositories, during which it accesses, indexes and organises a view of your repositories (including all historical items) back to the earliest commits. If you are evaluating Crucible, we suggest that you index a single project, so you can use Crucible as soon as possible. If you choose to index your entire repository, be aware that this can take a long time (possibly days) for massive or complex repositories and can be more complex to set up (especially for Subversion). The basic process is slightly different for each SCM type.

11. Stop Crucible (optional)

In a command prompt, change directory to <FishEye home directory> and run this:

```
bin\stop.bat
```

Running Crucible as a Windows service

Crucible can be run as a service under Microsoft Windows using a Java Service Wrapper.

The service wrapper provides the following benefits:

- Allows Crucible, which is a Java application, to be run as a Windows Service.
- No need for a user to be logged on to the system at all times, or for a command prompt to be open and running on the desktop to be able to run Crucible.
- The ability to run Crucible in the background as a service, for improved convenience, system performance and security.
- Crucible is launched automatically on system startup and does not require that a user be logged in.
- Users are not able to stop, start, or otherwise tamper with Crucible unless they are an administrator.
- Provides advanced failover, error recovery, and analysis features to make sure that Crucible has the maximum possible uptime.

⚠️ Please note that:

- This page should be read in conjunction with Installing Crucible on Windows.
- You should use 32-bit Java to run the service wrapper provided via the link in the install instructions below, even on a 64-bit machine.
- You should use the Java JDK, rather than the JRE, to take advantage of the `-server` parameter, provided in the Wrapper configuration of `wrapper.zip`, which enables the Java HotSpot(TM) Server VM. See the note below for details.

### Installing the Java Service Wrapper

To install the Java Service Wrapper on Windows:

1. Download `wrapper.zip` from [here](#).
2. Unzip the wrapper zip file into your `<FishEye home directory>` (that is, the directory into which Crucible was originally installed). Note, the resulting folder structure should be `<FishEye home directory>/wrapper, <FishEye home directory>/wrapper/bin, etc and NOT `<FishEye home directory>/wrapper/wrapper, <FishEye home directory>/wrapper/wrapper/bin. The location of the wrapper directory is important.
3. Tell the wrapper where to find the Java JDK by editing the `<FishEye home directory>/wrapper/conf/wrapper.conf` file, replacing this:

```ini
# Java Application
wrapper.java.command=java
```

with the following, and comment out the option you don't wish to use:

```ini
# Java Application

# Option 1: If you have JAVA_HOME defined in your Windows system environment variables, then you can use:
wrapper.java.command=%JAVA_HOME%/bin/java

# Option 2: If you have multiple JDKs installed, and you don't want to use a Windows environment variable to specify which one to use, provide the absolute path to where the JDK is installed (e.g. C:/Java/jdk1.7.0_05/bin/java):
wrapper.java.command=C:/<path to Java location>/bin/java
```

To get confirmation in the wrapper log that the wrapper is using the correct Java JDK, add the following lines to the `wrapper.conf` file:

```ini
# To get confirmation in the wrapper log that the wrapper is using the correct Java JDK, add the following lines to the
# wrapper.conf file:

wrapper.java.command=java
```

### Related pages:

- [Installing Crucible on Windows](#)
# Tell the Wrapper to log the full generated Java command line.
wrapper.java.command.loglevel=INF

You can find the logs at `<FishEye home directory>\var\log\wrapper.log`.

4. Set the FISHEYE_INST environment variable (and other Crucible-specific environment variables) in the `<FishEye home directory>\wrapper\conf\wrapper.conf` file, following the instructions below.
5. Install Crucible as a service as follows:
   a. Open an Administrator command prompt by searching for ‘Command prompt’ in the Windows Start menu, right-clicking on Command Prompt and then choosing Run as administrator.
   b. Change directory to `<FishEye home directory>\wrapper\` and run `bin\Fisheye-Install-NTService.bat`. If you run into any problems starting the wrapper, you’ll find its logs in `<FishEye home directory>\var\log\wrapper.log`.
6. Start the Crucible service (which has the name ‘Fisheye’) from the Windows Control Panel; you can search in the Start menu for ‘services’, and in the list of services, right-click on the ‘Fisheye’ item and choose Start. You can also stop the Crucible service in this way.

Please note that:

- If you make changes to the wrapper.conf file, having already started the service, you need to stop and then restart the service for it to make use of the changed configuration.
- If in future you move the FishEye home directory, you will need to uninstall (using `Fisheye-Uninstall-NTService.bat`) and then reinstall the Crucible service.

## Setting Crucible environment variables for Windows Services

Please note, that if you run Crucible as a Windows service, any Crucible-specific environment variables must be set in your `<FishEye home directory>\wrapper\conf\wrapper.conf` file.

If you run into any problems starting the wrapper, you’ll find its logs in `<FishEye home directory>\var\log\wrapper.log`.

If there are other Java parameters you wish to add, then you will need to add them under the additional parameters section, e.g.

```java
# JDK Additional Parameters for jmx
wrapper.java.additional.4=-Dcom.sun.management.jmxremote
wrapper.java.additional.5=-Dcom.sun.management.jmxremote.port=4242
wrapper.java.additional.6=-Dcom.sun.management.jmxremote.authenticate=false
wrapper.java.additional.7=-Dcom.sun.management.jmxremote.ssl=false
wrapper.java.additional.8=-Dcom.sun.management.jmxremote.authenticate=false
wrapper.java.additional.9=-Dcom.sun.management.jmxremote.password.file=/wrapper/jmxremote.password
wrapper.java.additional.10=-Dwrapper.mbean.name="wrapper:type=Java Service Wrapper Control"
```

To add the FISHEYE_INST environment variable, the Java MaxPermSize parameter, or the -Xrs options, use
the following:

```
wrapper.java.additional.11=-Dfisheye.inst="c:/path/to/FISHEYE_INST"
wrapper.java.additional.12=-XX:MaxPermSize=128m
wrapper.java.additional.13=-Xrs
```

Note that the the -Xrs options should be used when running Crucible as a service under Windows to prevent the JVM closing when an interactive user logs out.

Your memory settings can also be found in this file:

```
# Initial Java Heap Size (in MB)
wrapper.java.initmemory=256

# Maximum Java Heap Size (in MB)
wrapper.java.maxmemory=1024
```

Increase these values if you have a large repository or expect to use more memory (init of 256, and a max of 1024 are the default values).

In Fisheye/Crucible 1.6.4 and higher, you can check the JVM input arguments by clicking **System info**, under 'System Settings' in the admin area.

**Troubleshooting**

**Extracting files from wrapper.zip**

Some customers have reported trouble running the wrapper. These can be avoided by:

- Uncompressing wrapper.zip with Winzip or WinRar rather than using the Extract All command in the Windows right-click contextual menu.
- If the wrapper.zip filename appears green instead of black in Windows Explorer, decrypt it, prior to unzipping its contents, by right-clicking on the file, choose Properties, click the Advanced button, then clear the Encrypt contents to secure data checkbox.

**Warning when using 64-bit Java JDK**

When using a 64-bit Java JDK with the wrapper obtained via the link in the install instructions above, you may see the following in the wrapper.log file:

```
WARNING - Unable to load the Wrapper's native library 'wrapper.dll'. The file is located on the path at the following location but could not be loaded:
C:\installs\service\fisheye28\wrapper\lib\wrapper.dll.

Please verify that the file is readable by the current user and that the file has not been corrupted in any way. System signals will not be handled correctly.
```

This is caused by using a 64-bit JDK (even on a 64-bit machine). Changing to a 32-bit version of the JDK will prevent this warning. Community Edition versions of the 64-bit Windows Java Service Wrapper are not currently available.

**Wrapper configuration and "-server" parameter**
Please note that the wrapper configuration provided above uses the `-server` parameter to enable the Java HotSpot(TM) Server VM. This feature is only available if you use the JDK. If you use the JRE you will likely get the following error in your logs:

```
INFO | jvm | 2010/12/20 18:19:28 | Error: missing 'server' JVM at 'C:\Program Files\Java\jre6\bin\server\jvm.dll'.
```

A common issue is that customers remove the `-server` parameter from the `wrapper.conf` file. Please note that if you do this, the wrapper script will ignore any of the following JVM parameters in the file unless you change the sequence to be in order, starting from `wrapper.java.additional.1`. This is an issue with the Wrapper application.

In this situation it’s best to install and run Fisheye/Crucible with the JDK to get all the advantages of the `-server` functionality. You also need to force the wrapper to use the JDK by specifying the path to the Java JDK in the `wrapper.conf` file, as described in the installation instructions above.

### Installing Crucible on Linux and Mac

Hey! We're going to install Crucible on a Linux box, or a Mac. There are a few steps involved, but we think you'll find it easy to follow along. If you already have FishEye installed, you should read [Upgrading from FishEye to Crucible](#) instead.

#### 1. Check supported platforms

Better check the [Supported Platforms](#) page first; it lists the application servers, databases, operating systems, web browsers and JDKs that we have tested Crucible with, and that we recommend.

Atlassian only officially supports Crucible running on x86 hardware and 64-bit derivatives of x86 hardware.

*Related pages:*
- Installing Crucible on Windows
- Starting to use Crucible
- Supported Platforms

#### 2. Check your version of Java

In a terminal, run this:

```
java -version
```

The version of Java should be **1.6.0** or higher.

*If you don’t see Java 1.6.0 or higher, then get Java...*

Download and install the Java Platform JDK from [Oracle’s website](#).

Now try running `java -version` again to check the installation. The version of Java should be **1.6.0** or higher.

#### 3. Check that the system can find Java

In a terminal, run this:
echo $JAVA_HOME

You should see a path like /System/Library/Frameworks/JavaVM.framework/Versions/CurrentJDK/Home/

If you don’t see a path to the Java location, then set JAVA_HOME...

<table>
<thead>
<tr>
<th>Linux</th>
<th>Mac</th>
</tr>
</thead>
<tbody>
<tr>
<td>Do either of the following:</td>
<td>Insert the following in your ~/.profile file:</td>
</tr>
</tbody>
</table>
| • If JAVA_HOME is not set, log in with ‘root’ level permissions and run: | JAVA_HOME="path/to/JAVA_HOME"
export JAVA_HOME

where path/to/JAVA_HOME may be like: /System/Library/Frameworks/JavaVM.framework/Versions/CurrentJDK/Home/

• If JAVA_HOME needs to be changed, open the /etc/environment file in a text editor and modify the value for JAVA_HOME to:

JAVA_HOME="path/to/JAVA_HOME"

It should look like:

JAVA_HOME=/System/Library/Frameworks/JavaVM.framework/Versions/CurrentJDK/Home/

Refresh your ~/.profile in the terminal and confirm that JAVA_HOME is set:

source ~/.profile
$JAVA_HOME/bin/java -version

You should see a version of Java that is 1.6.0 or higher, like this:

java version "1.6.0_24"

4. Create a dedicated Crucible user (recommended)

For production installations, we recommend that you create a new dedicated user that will run Crucible on your system. This user:

• Should not have admin privileges.
• Should be a non-privileged user with read, write and execute access on the Crucible home (install) directory and instance (data) directory. These directories are described below.
• Should only have read access to your repositories.

If you created a dedicated Crucible user, ensure you are logged in as this user to complete the remaining instructions.

5. Now it’s time to get Crucible
Download Crucible from the Atlassian download site.

Extract the downloaded file to an install location:

- Folder names in the path to your Crucible executable should not have spaces in them. The path to the extracted directory is referred to as the `<FishEye home directory>` in these instructions, for consistency with the FishEye documentation.
- If you expect to have a large number of users for this Crucible installation, and Crucible will be connected to an external database, consider installing Crucible on a different server from the one running the external database, for improved performance.

6. Tell Crucible where to store your data

The Crucible instance directory is where your Crucible data is stored.

1. Create your Crucible instance directory.
2. Tell Crucible where you created it by adding a FISHEYE_INST environment variable as follows:

<table>
<thead>
<tr>
<th>Linux</th>
<th>Mac</th>
</tr>
</thead>
<tbody>
<tr>
<td>Open the <code>/etc/environment</code> file in a text editor and insert:</td>
<td></td>
</tr>
<tr>
<td>FISHEYE_INST=&quot;path/to/&lt;Crucible instance directory&gt;&quot;</td>
<td></td>
</tr>
<tr>
<td>Open the <code>~/.profile</code> file for the current user in a text editor and insert:</td>
<td></td>
</tr>
<tr>
<td>FISHEYE_INST=&quot;path/to/&lt;Crucible instance directory&gt;&quot;</td>
<td></td>
</tr>
<tr>
<td>export FISHEYE_INST</td>
<td></td>
</tr>
</tbody>
</table>

3. Now copy the newly extracted `<FishEye home directory>/config.xml` file to the root of your new FishEye instance directory.

⚠️ You should not locate your Crucible instance directory inside the `<FishEye home directory>` — they should be entirely separate locations. If you do put the instance directory in the `<FishEye home directory>` it will be overwritten, and lost, when Crucible gets upgraded. And by the way, you'll need separate Crucible instance directories if you want to run multiple copies of Crucible.

If you have a large number of repositories, we recommend you increase the default number of files that FishEye is allowed to open. See the following knowledge base article for more info: Subversion Indexer Paused with "Too many open files" Error.

7. Start Crucible!

In a terminal, change directory to `<FishEye home directory>` and run this:

```
bin/start.sh
```

After a few moments, in a web browser on the same machine, go to http://localhost:8060/ (or, from another machine, type http://hostname:8060/, where hostname is the name of the machine where you extracted Crucible).

Enter your license, then an admin password, to finish the setup.

You can postpone setting up JIRA integration until later if you wish; see Configuring JIRA integration in the Setup Wizard.
8. Connect to an external database (recommended)

If you intend to use this Crucible installation in a production environment, it is highly recommended that you use one of the supported external databases. See Migrating to an external database.

If you are evaluating Crucible, or don’t wish to do this now, Crucible will happily use its embedded database, and you can easily migrate later.

9. Set up your mail server

Configure the Crucible email server so that users can get notifications from Crucible. See Configuring SMTP.

10. Add users and repositories

Now is the time to set up your users in Crucible, and to tell Crucible about any existing repositories you have. Please read Starting to use Crucible for the details.

Crucible will perform an initial index of your repositories, during which it accesses, indexes and organizes a view of your repositories (including all historical items) back to the earliest commits. If you are evaluating Crucible, we suggest that you index a single project, so you can use Crucible as soon as possible. If you choose to index your entire repository, be aware that this can take a long time (possibly days) for massive or complex repositories and can be more complex to set up (especially for Subversion). The basic process is slightly different for each SCM type.

11. Stop Crucible (optional)

In a terminal, change directory to `<FishEye home directory>` and run this:

```
bin/stop.sh
```

Configuring JIRA Integration in the Setup Wizard

This page describes the ‘Connect to JIRA’ screen of the Crucible setup wizard.

You can connect your application to a JIRA server, to manage your users via JIRA and share information with JIRA. When you are installing the application, the setup wizard gives you the opportunity to configure the JIRA connection automatically. This is a quick way of setting up your JIRA integration with the most common options.

You can also configure the JIRA connections via the application administration screens. In that case, you will need to set up connections individually. There are two parts to the integration process:

- A peer-to-peer link between JIRA and the application for sharing information and facilitating integration features. This link is set up via Application Links.
- A client-server link between the application and JIRA for delegating user and group management to your JIRA server.

Requirements: You need JIRA 4.3 or later.

On this page:
- Connecting to JIRA in the Setup Wizard
- Troubleshooting
- Notes
Connecting to JIRA in the Setup Wizard

To configure JIRA integration while running the Crucible setup wizard:

1. Enter the following information on the ‘Connect to JIRA’ step of the setup wizard:
   - **JIRA Base URL** – The web address of your JIRA server. Examples:
     - http://www.example.com:8080/jira/
     - http://jira.example.com
   - **Admin Username** and **Admin Password** – The credentials of a user with the ‘JIRA System Administrators’ global permission in JIRA.
   - **FishEye/Crucible Base URL** – Click ‘Advanced Options’ to see this field. JIRA will use this URL to access your FishEye/Crucible server. The URL you give here will override the base URL specified in your FishEye/Crucible administration console, for the purposes of the JIRA connection.
   - **Groups to synchronize** – Click ‘Advanced Options’ to see this field. Select at least one JIRA group to synchronize. The default group is `jira-users`. JIRA will synchronize all changes in the user information on a regular basis. The default synchronization interval is 1 hour.
   - **Admin Groups** – Click ‘Advanced Options’ to see this field. Specify a JIRA group whose members should have administrative access to FishEye/Crucible. The default group is `jira-administrators`.

2. Click the ‘Connect to JIRA’ button.
3. Finish the setup process.
4. Configure the following setting in JIRA: **Allow remote API access**.

Screenshot: Connecting to JIRA in the FishEye/Crucible setup wizard
Troubleshooting

**Click to see troubleshooting information...**

This section describes the possible problems that may occur when integrating your application with JIRA via the setup wizard, and the solutions for each problem.

<table>
<thead>
<tr>
<th>Symptom</th>
<th>Cause</th>
<th>Solution</th>
</tr>
</thead>
</table>
| The setup wizard displays one of the following error messages:  
  - Failed to create application link from JIRA server at <URL> to this <application> server at <URL>.  
  - Failed to create application link from this <application> server at <URL> to JIRA server at <URL>.  
  - Failed to authenticate application link from JIRA server at <URL> to this <application> server at <URL>.  
  - Failed to authenticate application link from <application> server at <URL> to this JIRA server at <URL>. | The setup wizard failed to complete registration of the peer-to-peer application link with JIRA. JIRA integration is only partially configured. | Remove the partial configuration if it exists, try the ‘Connect to JIRA’ step again, and then continue with the setup. Detailed instructions are below. |
| The setup wizard displays one of the following error messages:  
  - Failed to register <application> configuration in JIRA for shared user management. Received invalid response from JIRA: <response>  
  - Failed to register <application> configuration in JIRA for shared user management. Received: <response> | The setup wizard failed to complete registration of the client-server link with JIRA for user management. The peer-to-peer link was successfully created, but integration is only partially configured. | Remove the partial configuration if it exists, try the ‘Connect to JIRA’ step again, and then continue with the setup. Detailed instructions are below. |
| The setup wizard displays the following error message:  
  - Error setting Crowd authentication | The setup wizard successfully established the peer-to-peer link with JIRA, but could not persist the client-server link for user management in your config.xml file. This may be caused by a problem in your environment, such as a full disk. | Please investigate and fix the problem that prevented the application from saving the configuration file to disk. Then remove the partial configuration if it exists, try the ‘Connect to JIRA’ step again, and then continue with the setup. Detailed instructions are below. |
<table>
<thead>
<tr>
<th>The setup wizard displays the following error message:</th>
<th>The setup wizard has completed the integration of your application with JIRA, but is unable to start synchronizing the JIRA users with your application.</th>
<th>Restart your application. You should then be able to continue with the setup wizard. If this solution does not work, please contact Atlassian Support.</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Error reloading Crowd authentication</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>The setup wizard displays the following error message:</th>
<th>The setup wizard has not completed the integration of your application with JIRA. The links are only partially configured. The problem occurred because there is already a user management configuration in JIRA for this &lt;application&gt; URL.</th>
<th>Remove the partial configuration if it exists, try the 'Connect to JIRA' step again, and then continue with the setup. Detailed instructions are below.</th>
</tr>
</thead>
<tbody>
<tr>
<td>• An error occurred: java.lang.IllegalStateException: Could not create the application in JIRA/Crowd (code: 500). Please refer to the logs for details.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>No users can log in after you have setup the application with JIRA integration.</th>
<th>Possible causes:</th>
<th>Go to JIRA and add some user names to the group.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>• There are no users in the group that you specified on the 'Connect to JIRA' screen.</td>
<td>• For FishEye: Go to the FishEye administration screens and specify at least one group to synchronize. The default is 'jira-users'.</td>
</tr>
<tr>
<td></td>
<td>• For FishEye: There are no groups specified in the 'groups to synchronize' section of your administration console.</td>
<td>• For Stash: Grant the Stash User permission to the relevant JIRA groups on the Stash Global permissions page.</td>
</tr>
<tr>
<td></td>
<td>• For Stash: You may not have granted any JIRA groups or users permissions to log in to Stash.</td>
<td></td>
</tr>
</tbody>
</table>

**Solution 1: Removing a Partial Configuration – The Easiest Way**

If the application's setup wizard fails part-way through setting up the JIRA integration, you may need to remove the partial configuration from JIRA before continuing with your application setup. Please follow the steps below.

Remove the partial configuration if it exists, try the 'Connect to JIRA' step again, and then continue with the setup wizard:

1. Log in to JIRA as a user with the 'JIRA System Administrators' global permission.
2. Click the 'Administration' link on the JIRA top navigation bar.
3. Remove the application link from JIRA, if it exists:
   a. Click 'Application Links' in the JIRA administration menu. The 'Configure Application Links' page will appear, showing the application links that have been set up.
   b. Look for a link to your application. It will have a base URL of the application linked to JIRA. For example:
      - If you want to remove a link between JIRA and FishEye, look for the one where the 'Application URL' matches the base URL of your FishEye server.
      - If you want to remove a link between JIRA and Confluence, look for the one where the 'Application URL' matches the base URL of your Confluence server.
      - If you want to remove a link between JIRA and Stash, look for the one where the 'Application URL' matches the base URL of your Stash server.
   c. Click the 'Delete' link next to the application link that you want to delete.
   d. A confirmation screen will appear. Click the 'Confirm' button to delete the application link.
4. Remove the user management configuration from JIRA, if it exists:
a. Go to the JIRA administration screen for configuring the applications that have been set up to use JIRA for user management:
   - In JIRA 4.3: Click ‘Other Applications’ in the ‘Users, Groups & Roles’ section of the JIRA administration screen.
   - In JIRA 4.4: Select 'Administration' > 'Users' > ‘JIRA User Server’.
b. Look for a link to your application. It will have a name matching this format:

   <Type> - <HostName> - <Application ID>

   For example:

   FishEye / Crucible - localhost - 92004b08-5657-3048-b5dc-f886e662ba15
   Or:

   Confluence - localhost - 92004b08-5657-3048-b5dc-f886e662ba15

   If you have multiple servers of the same type running on the same host, you will need to match the application ID of your application with the one shown in JIRA. To find the application ID:
   - Go to the following URL in your browser:
   
   <baseUrl>/rest/applinks/1.0/manifest

   Replace <baseUrl> with the base URL of your application. For example:

   http://localhost:8060/rest/applinks/1.0/manifest
   - The application links manifest will appear. Check the application ID in the <id> element.
c. In JIRA, click ‘Delete’ next to the application that you want to remove.

5. Go back to the setup wizard and try the ‘Connect to JIRA’ step again.

Solution 2: Removing a Partial Configuration – The Longer Way

If solution 1 above does not work, you may need to remove the partial configuration and then add the full integration manually. Please follow these steps:

1. Skip the 'Connect to JIRA' step and continue with the setup wizard, to complete the initial configuration of the application.
2. Log in to JIRA as a user with the ‘JIRA System Administrators’ global permission.
3. Click the ‘Administration’ link on the JIRA top navigation bar.
4. Remove the application link from JIRA, if it exists:
   a. Click ‘Application Links’ in the JIRA administration menu. The 'Configure Application Links' page will appear, showing the application links that have been set up.
   b. Look for a link to your application. It will have a base URL of the application linked to JIRA. For example:
      - If you want to remove a link between JIRA and FishEye, look for the one where the 'Application Links' element.

   c. In JIRA, click ‘Delete’ next to the application that you want to remove.

   5. Go back to the setup wizard and try the ‘Connect to JIRA’ step again.
Documentation for Crucible 2.8

If you want to remove a link between JIRA and Confluence, look for the one where the ‘Application URL’ matches the base URL of your Confluence server.

If you want to remove a link between JIRA and Stash, look for the one where the ‘Application URL’ matches the base URL of your Stash server.

c. Click the ‘Delete’ link next to the application link that you want to delete.

d. A confirmation screen will appear. Click the ‘Confirm’ button to delete the application link.

5. Remove the user management configuration from JIRA, if it exists:

a. Go to the JIRA administration screen for configuring the applications that have been set up to use JIRA for user management:

   • In JIRA 4.3: Click ‘Other Applications’ in the ‘Users, Groups & Roles’ section of the JIRA administration screen.
   
   • In JIRA 4.4: Select ‘Administration’ > ‘Users’ > ‘JIRA User Server’.

b. Look for a link to your application. It will have a name matching this format:

   ```
   <Type> - <HostName> - <Application ID>
   ```

   For example:

   ```
   FishEye / Crucible - localhost - 92004b08-5657-3048-b5dc-f886e662ba15
   ```

   Or:

   ```
   Confluence - localhost - 92004b08-5657-3048-b5dc-f886e662ba15
   ```

   If you have multiple servers of the same type running on the same host, you will need to match the application ID of your application with the one shown in JIRA. To find the application ID:

   • Go to the following URL in your browser:

     ```
     <baseUrl>/rest/applinks/1.0/manifest
     ```

     Replace `<baseUrl>` with the base URL of your application.

     For example:

     ```
     http://localhost:8060/rest/applinks/1.0/manifest
     ```

   • The application links manifest will appear. Check the application ID in the `<id>` element.

   c. In JIRA, click ‘Delete’ next to the application that you want to remove.

6. Add the application link in JIRA again, so that you now have a two-way trusted link between JIRA and your application:

   a. Click ‘Add Application Link’. Step 1 of the link wizard will appear.

   b. Enter the server URL of the application that you want to link to (the ‘remote application’).

   c. Click the ‘Next’ button.

   d. Enter the following information:

     • ‘Create a link back to this server’ – Tick this check box to add a two-way link between the two applications.
     
     • ‘Username’ and ‘Password’ – Enter the credentials for a username that has administrator

Created by Atlassian in 2012. Licensed under a Creative Commons Attribution 2.5 Australia License.
access to the remote application.

Note: These credentials are only used to authenticate you to the remote application, so that Application Links can make the changes required for the new link. The credentials are not saved.

• ‘Reciprocal Link URL’ – The URL you give here will override the base URL specified in your remote application's administration console, for the purposes of the application links connection. Application Links will use this URL to access the remote application.

e. Click the 'Next' button.

f. Enter the information required to configure authentication for your application link:
   • ‘The servers have the same set of users’ – Tick this check box, because the users are the same in both applications.
   • ‘These servers fully trust each other’ – Tick this check box, because you trust the code in both applications and are sure both applications will maintain the security of their private keys.

For more information about configuring authentication, see Configuring Authentication for an Application Link.

g. Click the 'Create' button to create the application link.

7. Configure a new connection for user management in JIRA:
   a. Go to the JIRA administration screen for configuring the applications that have been set up to use JIRA for user management:
      • In JIRA 4.3: Click 'Other Applications' in the 'Users, Groups & Roles' section of the JIRA administration screen.
      • In JIRA 4.4: Select 'Administration' > 'Users' > 'JIRA User Server'.
   b. Add an application.
   c. Enter the application name and password that your application will use when accessing JIRA.
   d. Enter the IP address or addresses of your application. Valid values are:
      • A full IP address, e.g. 192.168.10.12.
      • A wildcard IP range, using CIDR notation, e.g. 192.168.10.1/16. For more information, see the introduction to CIDR notation on Wikipedia and RFC 4632.
   e. Save the new application.

8. Set up the JIRA user directory in the application.
   • For Confluence:
      a. Go to the Confluence Administration Console.
      b. Click 'User Directories' in the left-hand panel.
      c. Add a directory and select type 'Atlassian JIRA'.
      d. Enter the following information:
         • Name – Enter the name of your JIRA server.
         • Server URL – Enter web address of your JIRA server. Examples:

http://www.example.com:8080/jira/  
http://jira.example.com

• Application name and Application password – Enter the values that you defined for Confluence in the settings on JIRA.

   e. Save the directory settings.
   f. Define the directory order by clicking the blue up- and down-arrows next to each directory on the 'User Directories' screen. For details see Connecting to Crowd or JIRA for User Management.

   • For FishEye/Crucible:
      a. Click Authentication (under 'Security Settings').
      b. Click Setup JIRA/Crowd authentication. Note, if LDAP authentication has already been set up, you will need to remove that before connecting to JIRA for user management.
      c. Make the following settings:
<table>
<thead>
<tr>
<th>Authenticate against</th>
<th>Select a JIRA instance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application name and password</td>
<td>Enter the values that you defined for your application in the settings on JIRA.</td>
</tr>
<tr>
<td>JIRA URL</td>
<td>The web address of your JIRA server. Examples:</td>
</tr>
</tbody>
</table>

```
http://www.example.com:8080/jira/
http://jira.example.com
```

<table>
<thead>
<tr>
<th>Auto-add</th>
<th>Select Create a FishEye user on successful login so that your JIRA users will be automatically added as a FishEye user when they first log in.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Periodically synchronise users with JIRA</td>
<td>Select Yes to ensure that JIRA will synchronize all changes in the user information on a regular basis. Change the value for Synchronise Period if required.</td>
</tr>
<tr>
<td>When Synchronisation Happens</td>
<td>Select an option depending on whether you want to allow changes to user attributes from within FishEye.</td>
</tr>
<tr>
<td>Single Sign On</td>
<td>Select Disabled. SSO is not available when using JIRA for user management and if enabled will make the integration fail.</td>
</tr>
</tbody>
</table>

d. Click Next and select at least one user group to be synchronised from JIRA. If necessary, you could create a new group in JIRA, such as 'fisheye-users', and select this group here.
e. Click Save.

- For Stash:
  a. Go to the Stash Administration Console.
  b. Click 'User Directories' in the left-hand panel.
  c. Add a directory and select type 'Atlassian JIRA'.
  d. Enter the following information:
     - **Name** – Enter the name of your JIRA server.
     - **Server URL**– Enter web address of your JIRA server. Examples:

```
http://www.example.com:8080/jira/
http://jira.example.com
```

- **Application name and Application password** – Enter the values that you defined for Stash in the settings on JIRA.
e. Save the directory settings.
f. Define the directory order by clicking the blue up- and down-arrows next to each directory on the 'User Directories' screen.
For details see Connecting to JIRA for user management.
Notes

When you connect to JIRA in the setup wizard, the setup procedure will configure Trusted Applications authentication for your application. Please be aware of the following security implications:

- Trusted applications are a potential security risk. When you configure Trusted Applications authentication, you are allowing one application to access another as any user. This allows all of the built-in security measures to be bypassed. Do not configure a trusted application unless you know that all code in the application you are trusting will behave itself at all times, and you are sure that the application will maintain the security of its private key.

Starting to use Crucible

This page will guide you though the basics of using Crucible. By the end of it you should know how to:

- Add a repository.
- Create a project.
- Create and perform reviews.

Assumptions

This page assumes that:

- You have installed and started the latest version of Crucible
- You are using a supported browser.

Install Crucible

Get Crucible running on your computer. See the details here:

- Installing Crucible on Windows
- Installing Crucible on Linux and Mac

Add a repository

In this section we’re going to add a repository to Crucible.

Click on Add Existing... in the Repositories listing of the Administration.
Choose the repository type and fill in the name and description.

In the repository configuration put the location of your repository. Fill in the authentication details if necessary.
Finally indicate whether or not you would like diff indexing should be turned on and if the repository should not be indexed right away then click Add to finish the process.

Your repository is now created in Crucible and the indexing should have started.

Create a project in Crucible

Crucible comes with a default project with the key CR but you will most likely have to create your own projects that will contain your reviews. This is achieved in a couple of steps.

Click on Add a new project in the Projects listing of the Administration.
Fill in the form with the default settings for the project and hit Save.

You should then be taken back to the Project listing where you can see your new project created.
Create a review

Now that you have your own project it is possible to create reviews in it.

⚠️ You need to be logged in to create reviews.

From the Dashboard click on Create review to open the review creation form.

Choose the project in which you want to create the review.
In the next screen click on **Browse Changeset** to see the list of changesets available to create the review.

Select the changesets that you would like to be reviewed and then click on **Edit Details** to add reviewers.
Once that you have chosen your reviewers and updated the review information you can click on **Start Review** to begin the review process.

The review is now created and the reviewers have been notified that a review is pending.
In order to close a review where you are the moderator you need to click on **Summarize** in the top right and then close the review from the modal window.

**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.

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 “three hats”, acting as review creator, moderator 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, moderator and author are pre-selected (for this example, the author should select himself as a moderator).

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 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.)
2. The Reviewer Comments on the Code

The reviewer will receive an email from Crucible (or an RSS feed update) 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/moderator 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/moderator is notified via email. The author/moderator clicks the link in the notification email, returning to the Review screen.

The author/moderator will then add any final comments, then click Summarise when finished. The Crucible Summarize Review screen opens.

Screenshot: Summarizing a review in Crucible

On the Crucible Summarize Review screen, the author/moderator enters an optional summary of the review's results, then clicks Close Review. 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, displaying the summary and archiving the completed review as read-only.

Screenshot: Viewing a closed review
If the author/moderator 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>Source Dropdown Menu</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>Projects Dropdown Menu</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>People Dropdown Menu</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>People Dropdown Menu</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:
   - Note, clicking any of the links below will redirect you to the ‘Reviews' tab on the Dashboard. You can click the ‘Reviews' tab at the top of the screen, if you need to return to this (e.g. to access the ‘Reports' tab).
   - ‘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>
</tbody>
</table>
| Activity           | Shows recent activity on the item. There are a number of sub-options here:  
|                   | * All Activity — The default view, showing commits, reviews and JIRA issues.  
|                   | * Commits — Shows commits in the activity stream.  
|                   | * Reviews — Shows review activity in the activity stream.  
|                   | * Scroll to Changeset — Opens the changeset ID specified in the text field (press Enter to carry out the action).  
|                   | * Filter — Applies constraints to the current activity stream.  
|                   | * Show Revisions — If this is selected, then changeset items are automatically expanded to show modified files.  
|                   | * Earlier Activity (Left Arrow icon) — Loads a page of earlier activity.  
|                   | * Later Activity (Right Arrow icon) — Loads a page of later activity. |

| 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. |

ℹ️ 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**

![Viewing all repositories](image)

**Screenshot: Viewing a repository**

![Viewing a repository](image)
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 (จอง) to hide any of the sections. Click the expand icon (.expand) to expand any collapsed sections.
- Click the 'i' icon (i) 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' states:</th>
</tr>
</thead>
<tbody>
<tr>
<td>'To Review' — Click to show all reviews where you are a reviewer and haven't yet completed your review work.</td>
<td></td>
</tr>
<tr>
<td>'To Summarize' — Click to show all reviews where you are a moderator and haven't yet summarised and closed the review.</td>
<td></td>
</tr>
<tr>
<td>'In Draft' — 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>
<td></td>
</tr>
<tr>
<td>'Require My Approval' — Click to show all reviews where you are a moderator and need to approve the review.</td>
<td></td>
</tr>
</tbody>
</table>
‘Outbox’ Click to show all reviews in ‘Out for Review’ and ‘Completed’ states:

- ‘Out for Review’ — Click to show all reviews that you are a participant of, that have review work that is yet to be completed by other reviewers.
- ‘Completed’ — Click to show all reviews that you are a participant of, and have been completed.

‘Archive’ Click to show all reviews in ‘Closed’ and ‘Abandoned’ states:

- ‘Closed’ — Click to show all reviews that you are a participant of, that have been summarised and closed.
- ‘Abandoned’ — Click to show all reviews that you are a participant of, that have been abandoned. You may wish to delete these reviews.

‘My Open Snippets’ Click to show all open snippets created by you.

‘My Snippets’ Click to show all snippets created by you.

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**
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**

---

*Created by Atlassian in 2012. Licensed under a Creative Commons Attribution 2.5 Australia License.*
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 Filter 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.

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>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
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-28189.

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 <strong>authors</strong>.</td>
</tr>
<tr>
<td>Moderator</td>
<td>Find reviews moderated by a particular <strong>moderator</strong>.</td>
</tr>
<tr>
<td>Creator</td>
<td>Find reviews created by a particular <strong>creator</strong>.</td>
</tr>
<tr>
<td>Reviewer</td>
<td>Find reviews that are reviewed by a particular <strong>reviewer</strong>. This will default to the user logged in.</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 <strong>all</strong>. To use any of the filters, choose <strong>any</strong>.</td>
</tr>
</tbody>
</table>

### Review state checkboxes

Check any of the review state checkboxes (e.g. 'Draft', 'Pending Approval') to filter for reviews in those states.

---

**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>Filter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project</td>
<td>Find comments on reviews under a particular project.</td>
</tr>
<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.

| **Review State** | Find comments on reviews that are in a particular state. See *Review State Filter* (above). |
| **Ranking** | Find [defects](#) have been given a particular ranking (e.g. 'Major', 'Minor'). |
| **Requires Re-Review** | Find [defects](#) that have been marked as requiring re-review (or not). |
| **Classification** | Find [defects](#) that have been given a particular classification (e.g. 'Missing', 'Ambiguous'). |
Screenshot above: Search Comment Filter Options

Displaying Defect Metric Charts for Comment Search Results

Once you have retrieved results for a review comment search, you can click 'Defect Metrics' in the left navigation pane to display defect classification charts.

Screenshot above: Comment Search Results

Notes

Related Topics

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**

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

![Projects](image)

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><strong>52 week commits volume</strong></td>
<td>This chart shows the amount of commits, shown by week over a one year period.</td>
</tr>
<tr>
<td><strong>Commits by day</strong></td>
<td>This chart shows the amount of commits, graphed by day over the past week.</td>
</tr>
<tr>
<td><strong>Commits by hour</strong></td>
<td>This chart shows the amount of commits, graphed by hours over the past day.</td>
</tr>
<tr>
<td><strong>Commit calendar</strong></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.

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.

ℹ️ 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.

**Screenshot: Enabling Changeset Comments in the Admin Screens**

![Changeset Discussions](image)

| Enabled: false (Reset to Repository Default) | Edit |

## 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 Diagram]

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.

<table>
<thead>
<tr>
<th>Screenshot: Crucible Patch Anchoring</th>
</tr>
</thead>
</table>

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.

ℹ️ 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**.

*Screenshot: 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.
2. Under ‘Activity’, click the **Source** tab.
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:
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 ‘%2F’ 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 (click to view larger images)**

Selecting the Files for the Review

![Step 1](image1.png) ![Step 2](image2.png)

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**

![Add Content menu](image)

**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**

![Browse Changesets](image)
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 changesets.
- '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**

![Iterative Reviews Screenshot](image)

**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
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]

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]

### 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**

*Is this variable name sensible?*

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**

Edwin Dawson says:

*Is this variable name sensible?*

Reply Edit Delete Post Draft

**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*

![](Complete.png)

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:
   4. In the 'To' field you can enter multiple email addresses, separated by commas.
   5. In the 'Recipients' field, you can type usernames from your Crucible instance to add them to the distribution list. You can also simply tick the 'Send to Review Participants' check-box to include all of the review's reviewers.
6. When you have finished the distribution list, click the 'Next' button.
7. The 'Recipients' page appears (see Screenshot 3 below). On that page:
8. In the 'To' field you can enter multiple email addresses, separated by commas.
9. When you have finished your message, click the 'Send' button.
10. The 'Status' page appears (see Screenshot 4 below), confirming that your email has been sent correctly.

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.

You can also link a JIRA issue to the review after the review is created. When viewing a review, the details pane of the review shows meta-information. One of the links in this area is titled ‘Linked JIRA Issue’ and then a suggested JIRA issue key. Click this link to associate that JIRA issue with this review.

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
your mouse over the JIRA issue key, an information window will show more information and controls relating to that JIRA issue.

Screenshot: The JIRA Hover information window

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](image)

**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%.

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.

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.
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.

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.
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

As the moderator, you can choose to summarize a review at any time.

ℹ️ Normally, we recommend that you wait for all reviewers to complete their reviews.

The reviews that the reviewers have completed will be in your 'To Summarize' menu on the Dashboard.

To summarize a review,

- Click the 'Summarize' button at the right of the screen.
- Optionally enter a summary of the review.
- If you have no further comments to add, click the 'Close Review' button; 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 moderator 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

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

- The moderator is the only participant who can add comments in ‘Summarize’ state. This gives the moderator the responsibility of the ‘last word’.
- Reviews in the ‘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,
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 '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
### Key Function

<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.</td>
</tr>
<tr>
<td></td>
<td><strong>Biggest Text</strong></td>
</tr>
<tr>
<td>h2.Bigger heading</td>
<td>Turns text into a heading at size 2.</td>
</tr>
<tr>
<td></td>
<td><strong>Bigger heading</strong></td>
</tr>
<tr>
<td>h3.Big heading</td>
<td>Turns text into a heading at size 3.</td>
</tr>
<tr>
<td></td>
<td><strong>Big heading</strong></td>
</tr>
<tr>
<td>h4.Normal heading</td>
<td>Turns text into a heading at size 4.</td>
</tr>
<tr>
<td></td>
<td><strong>Normal heading</strong></td>
</tr>
</tbody>
</table>
h5.Small heading
Turns text into a heading at size 5.

h6.Smallest heading
Turns text into a heading at size 6.

Text Effects
Text effects are used to change the formatting of words and sentences.

<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 <strong>underlined</strong>.</td>
</tr>
<tr>
<td>??citation??</td>
<td>Makes text appear in <em>citation form</em>.</td>
</tr>
<tr>
<td>-strikethrough-</td>
<td>Makes text appear <strong>struck through</strong>.</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>
</tbody>
</table>

Placing double curly-brackets around text makes it appear monospaced.

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>
</tbody>
</table>
| \\
| Creates a line break. |
| ----
| Creates a horizontal ruler. |
| ---
| Produces em dash — symbol. |
| --
| Produces en dash – symbol. |

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]<a href="http://atlassian.com">http://atlassian.com</a></td>
<td>Creates a link to an external resource, special characters that come after the URL and are not part of it must be separated with a space. External links are denoted with an arrow icon.</td>
</tr>
</tbody>
</table>

Examples:

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

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).
**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 |
| # A numbered list  
# Second item  
## indented item 1  
## indented item 2 | 1. A numbered list  
2. Second item  
  a. indented item 1  
  b. 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><strong>My Title</strong></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
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. <code>{</code>)</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.
On this page:

- Overview of Crucible Gadgets
- Gadget Configuration 1: Add JIRA to FishEye as an OAuth consumer
- Gadget Configuration 2: Add Gadget to your Application's Gadget Directory
- Gadget Configuration 3: Add Gadget to the Application's Dashboard

**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.</td>
</tr>
<tr>
<td></td>
<td>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.</td>
</tr>
<tr>
<td></td>
<td>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>
| **'Overdue Reviews Gadget'** | This gadget shows you reviews that are yet to be completed in the project, across all authors. This is useful for managers or team leads.  

The URL for this gadget is as follows:  

http://HOSTNAME:8060/rest/gadgets/1.0/g/com.atlassian.fecru.fecru-gadgets-plugin:overdueReviews/gadgets/overdueReviews.xml  

In this example, HOSTNAME:8060 is the hostname of your Crucible instance. |
|---|---|
| **Review Coverage** | This gadget shows content from the innovative Review Coverage report, letting you investigate how much of your codebase has been under code review.  

The URL for this gadget is as follows:  

http://HOSTNAME:8060/rest/gadgets/1.0/g/com.atlassian.fisheye.review-coverage-report/gadgets/recent-changesets.xml  

In this example, HOSTNAME:8060 is the hostname of your Crucible instance. |

Configuring gadgets is a three phase process. Firstly, you add your JIRA instance as an OAuth consumer. Secondly, you'll add the gadget to the destination application, then finally you add the gadget to the application's dashboard. In our example, we will show how to configure the Crucible gadgets for use in JIRA.

**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**

![OAuth Administration Screen](image)

OAuth is a protocol used when one web application needs to identify itself to another. When authorization is done via OAuth, users can log in to one application (the service provider) and access the application's information from another application (the consumer) without giving their password to the second application. Typically, the consumer accesses the data via gadgets that support OAuth for authorization.
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**

<table>
<thead>
<tr>
<th>Consumers</th>
<th>Consumer Info</th>
<th>Service Providers</th>
</tr>
</thead>
</table>

An OAuth consumer is a web site or application that has permission to access FishEye’s data on behalf of the logged-in user. Typically, the consumer accesses the data via gadgets that support OAuth for authorization. Learn more.

Enter the consumer’s base URL so we can retrieve its information automatically for you.

**Consumer Base URL**: http://jira.atlassian.com

*Use base URLs for Atlassian applications only. For example, "http://jira.mycompany.com".*

![Add](image)

**Return to Consumer List**

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**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Atlassian JIRA</td>
<td>Atlassian JIRA at <a href="http://jira.atlassian.com">http://jira.atlassian.com</a></td>
<td>Edit</td>
</tr>
</tbody>
</table>

**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](https://jira.atlassian.com) 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](https://jira.atlassian.com) 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](https://confluence.atlassian.com/pages/overview) 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>Summarize</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
Crucible.

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)

#### 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.

<i>This feature does not support editing fields on the issue, only setting the *resolution* field if required by the transition.</i>

#### Related Links

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

### Administering Crucible

Once you have installed and configured Crucible, you can access the **Administration** pages at the following address (where 'HOSTNAME' is the name of the server where you installed Crucible).
The 'Admin Menu' allows you to administer your Crucible instance and manage your repositories, users and back-end settings. For information on administering FishEye, please refer to the FishEye documentation.

Topics

- Crucible and FishEye
- Creating a User
- Administering projects
- Configuring Repositories
- Customising the Welcome Message
- Customising email notifications
- Setting up Users and Security
- Customising the Defect Classifications
- Creating a Permission Scheme
- Configuring User Managed Mappings
- Enabling Access Logging in Crucible
- Backing up and restoring Crucible data
- Best Practices for Crucible Configuration
- Managing plugins
- Migrating to an External Database
- Integrating Crucible with Other Applications

Crucible and FishEye

This page gives an overview of the joint installation of Crucible and FishEye. Both Crucible and FishEye are Atlassian products.

- FishEye allows you to extract information from your source code repository and display it in sophisticated reports.
- Crucible allows you to request, perform and manage code reviews.
- Both of these products can run in isolation. If you are using Subversion, Git, Mercurial, CVS or Perforce you can significantly enhance your Crucible experience by also using FishEye.
  - See What happens if I decide to stop using FishEye with Crucible? to learn more.

⚠ Your Crucible installation package includes the files required for FishEye

If you use FishEye and Crucible together, they run as one instance.

Purchasing and installing Crucible/FishEye

- If you install Crucible, there is no need to do a separate installation of FishEye.
- Upgrading Crucible to also use FishEye requires only a simple licence change in the admin screens.
- When upgrading to Crucible when you have an existing FishEye installation, you can either keep the original FishEye installation or install Crucible and FishEye as a fresh install. Refer to the guide on upgrading from FishEye to Crucible.

The <FishEye home directory> and FISHEYE_INST

Throughout the Crucible documentation, references are made to the <FishEye home directory>, which refers to the location of the FishEye application. Because most Crucible users also run FishEye, we use a single value for the sake of simplicity and consistency with the FishEye documentation.
Crucible also makes use of this FishEye environment variable:

- FISHEYE_INST – the location of the FishEye data.

Refer to the FishEye documentation for more about the environment variables.

**Installing the Crucible binary files**

See the following for Crucible install instructions:

- [Installing FishEye on Windows](#)
- [Installing FishEye on Linux and Mac](#)

**Setting up a repository for use with stand-alone Crucible**

When accessing repositories from stand-alone Crucible, the SCM client interface will access the repositories on demand. This requires that you configure the plugin.

For complete instructions, see [Configuring Repositories](#).

**Setting up a Repository for use with FishEye and Crucible**

- If you intend to use Crucible and FishEye with Subversion, please read [Supported platforms](#), Subversion client setup, and granting permission to FishEye to scan your repository.
- If you intend to use Crucible and FishEye with Git, please read [Supported platforms](#) and Git Client setup.
- If you intend to use Crucible and FishEye with Perforce, please read [Supported platforms](#) and Perforce Client setup.
- If you intend to use Crucible and FishEye with CVS, please read [Supported platforms](#) and CVS Client setup.
- If you intend to use Crucible and FishEye with Mercurial, please read [Supported platforms](#) and Mercurial Client setup.

**Detailed Documentation**

You can find more information in:

- Crucible [Getting Started](#)
- FishEye [Getting Started](#)

**Creating a User**

Some user management functions are identical between Crucible and FishEye, Crucible's sister application. See more user management documentation in the FishEye documentation.

There are two types of user accounts:

- 'Built-in' user accounts — these are stored in the application's local database.
- 'External' user accounts — these are stored in an external directory (e.g. LDAP), if any are configured.

See [Configuring External Authentication Sources](#).

**Note regarding external directories:**

- New users can only be added if they already exist in the external directory. Your external directory will not be modified.
- If you have enabled 'auto-add' for your external directory, users who don't exist locally will be automatically added the first time they log in.

To add a new user:
1. Click 'Users' on the 'Admin Menu'.
2. The 'User Browser' screen will be displayed (see screenshot below). Click the 'Add User' button at the bottom of the screen.
3. The 'Add new user' screen will be displayed.
4. In the 'Username' field, type the user's login name. You can use the following characters:
   - letters and numbers
   - hyphen ('-')
   - underscore ('_')
   - 'at' sign (@)
5. In the 'Display name' field, type the user's display-name.
6. (Optional) In the 'Email' field, type the user's email address. This address is where the user will receive notifications.
7. In the 'Auth Type' field, select either 'Built-in' or the name of the appropriate external directory where the user will be stored.
8. (For built-in users only) In the 'Password' and 'Confirm Password' fields, type the user's password. The user can easily change their own password later.
9. Click the 'Add' button.

Screenshot: User Browser

Administering projects

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'.

To view your projects:

1. From the 'Admin Menu', click 'Project List'.
2. The 'Projects List' page will be displayed.
   - Click the 'Create a New Project' link, which appears at the bottom of the list of existing projects, to
create a new project. See Creating a Project.
- Click the 'Edit' link next to an existing project's name, to edit the project. See Editing a Project.
- Click the 'Delete' link next to an existing project's name, to edit the project. See Deleting a Project.

**Creating a Project**

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'.

**To create a new project:**

1. Click the menu labelled with your user name in the the FishEye/Crucible header, and click the 'Administration' option. You will need to be logged in as an administrator to see this link. The FishEye/Crucible administration console will be displayed.
2. Click 'Projects' link in the left menu. The 'Projects' page will be displayed showing all of the projects that you have set up in Crucible.
3. Click the 'Create a New Project' link, at the bottom of the list of your existing projects.
4. The 'Edit Project' page will be displayed. Complete the fields on this page:
   - 'Name' — Type a short phrase that describes your project.
   - 'Key' — Type a few characters to uniquely identify your project. This key must consist of alphabetic and/or numeric characters and hyphens only.
   - 'Default Repository' — Select the repository which contains source code relating to this project.
     - This repository is the one that will be searched by default when you add files to a review.
   - 'Store the contents of files in reviews' — Tick this checkbox if you want to store the contents of files included in reviews, in the reviews themselves.
   - 'Permission Scheme' — Select the relevant permission scheme for this project. (A permission scheme controls who can perform particular actions, e.g. 'Create Review'.)
   - 'Enable the Moderator role for this project' — Tick this checkbox if you want enable the moderator role for this project.
   - 'Default Moderator' — Type the name of the person who will appear by default in the 'Moderator' field when you create a new review; or leave this field blank to force the review's creator to choose a moderator.
   - 'By default, allow anyone to join reviews after creation' — Tick this checkbox, if you want to allow anyone to join reviews after creation. See Adding Reviewers.
   - (Optional) Under 'Default Reviewers', select the people to whom new reviews in this project will be assigned by default:
     - 'Users' — Type the name(s) of individual users to whom new reviews will be assigned by default.
     - 'Groups' — Type the name(s) of groups to whose members new reviews will be assigned by default.
   - (Optional) Under Allowed Review Participants, select who will be allowed to have a role (i.e. be an author/creator/moderator/reviewer) in this project's reviews:
     - 'Users' — Type the name(s) of individual users who will be eligible to be authors/creators/moderators/reviewers for reviews in this project*.  
     - 'Groups' — Type the name(s) of groups whose members will be eligible to be authors/creators/moderators/reviewers for reviews in this project*.  
     * These users will be the only ones whose names appear when a review is assigned.
   - 'Default duration in week days' — Enter a value for the number of working days that you want the review to run for. Simply type the number of days into the text entry field marked 'Default
Setting Crucible to Store all Revisions

When creating a project or editing a project’s properties, you can set Crucible to save all revisions that are associated with a review to Crucible’s database. This allows you to be able to view that file content whether or not the repository is online or accessible to Crucible. It also creates an enhanced audit trail should you require it, saving the review content regardless of whether or not it is deleted or lost from the repository.

Note that the storage of revisions must be set per-project. Also, the storage only applies to reviews created after Revision Storage is enabled. This means that for existing projects, pre-existing reviews will not be stored unless you look at them again after Revision Storage is enabled.

Enabling Revision Storage on a new project

To enable Revision Storing on a new project:

- Click the 'Default Review Objectives' — Enter review objectives that will be applied to all new reviews.
- Click the 'Preview' button to preview the text entered in this field.
- Click the 'Save' button to create your new project.
1. When creating a new project, you have the option to turn on revision storing on the 'Create New Project' page.
2. Under 'Default Content Review Repository', Click the checkbox labelled 'Store the contents of files in reviews'.
3. Click 'Save' to finish.

Enabling Revision Storage on an existing project

To enable Revision Storing on an existing project:

1. From the 'Admin' screen, click 'Projects' from the left navigation bar.
2. Click 'Edit' next to the desired project.
3. Under 'Default Content Review Repository', Click the checkbox labelled 'Store the contents of files in reviews'.
4. Click 'Save' to finish.

Deleting a Project

Admin users can delete projects under Crucible. To do this, follow the instructions below.

Before you begin:

- By default, Crucible contains one project. This default project has the key 'CR' and the name 'Default Project'. This project cannot be deleted.
- Deleted projects cannot be recovered.

To delete a project from the Project List:

1. From the 'Admin Menu', click 'Project List'.
2. A list of projects appears. With care, click the 'Delete' link situated to the right of the project you wish to remove. If empty, the project instantly disappears.
3. If the project contains reviews, you will be prompted to either delete all reviews in the project, or move them into the default project.

Editing a Project
Once projects are created, you can return to the project settings page to change the defaults such as repository, moderator, allowed reviewers, allowed groups and permissions.

**To edit project settings:**

1. From the 'Admin Menu', click 'Project List'.
2. The list of projects will be displayed. Click the 'Edit' link for the desired project, which appears to the right of the existing project name.
3. The 'Edit Project' page will be displayed. You can now adjust any of the given settings as desired.
4. In the 'Identification' section, you can change the plain language name (as displayed in the Crucible interface) and the project key (used when giving reviews their unique code names).
5. In the 'Default Review Content Repository' field, you can adjust the repository which contains source code relating to this project.
   - This repository is the one that will be searched by default when you add files to a review.
   - The check box here labelled 'Store the contents of files in reviews' will cause the source files under review to be stored in the Crucible database along with the comments and review data. This will retain a copy of all the source files that go under review even in the event of disconnecting the repository from Crucible.
6. In the 'Default Moderator' field, you can adjust the name of the person who will appear by default in the 'Moderator' field when you create a new review; or leave this field blank to force the review's creator to choose a moderator.
   - You can also tick the option 'Disable Moderator' to have reviews run by the author only. See Enabling the Moderator Role for more information.
7. *(Optional)* Under 'Default Reviewers', you can adjust the people to whom new reviews in this project will be assigned by default:
   - Select the 'Let allowed review participants join a review' check-box if you wish to determine the default for the 'Allow anyone to join' option on the 'Adding Reviewers' screen.
   - In the 'Users' field, you can adjust the name(s) of individual users to whom new reviews will be assigned by default.
   - In the 'Groups' field, you can adjust the name(s) of groups to whose members new reviews will be assigned by default.
8. *(Optional)* Under Allowed Review Participants', you can adjust who will be allowed to have a role (i.e. be an author/creator/moderator/reviewer) in this project's reviews:
   - In the 'Users' field, you can adjust the list of individual users who will be eligible to be authors/creators/moderators/reviewers for reviews in this project*.
   - In the 'Groups' field, you can adjust the list of groups whose members will be eligible to be authors/creators/moderators/reviewers for reviews in this project*.
   - These users will be the only ones whose names appear when a review is assigned.
9. In the 'Permission Scheme' drop-down list, you can adjust the relevant permission scheme for this project. (A permission scheme controls who can perform particular actions, e.g. 'Create Review'.)
10. In the 'Review Duration' section, you can define the default length of time (in week days) for reviews in this project.
11. In the 'Default Review Objectives Section', you can define some text that will appear by default in the Review Objectives field of each new review. This text can be edited, as any text contained in the Review Objectives text box can.

*These users will be the only ones whose names appear when a review is assigned.*

Screenshot: The Edit Project screen in Crucible
Enabling the Moderator Role

This page contains instructions on how to enable or disable the moderator role for all reviews under a given Crucible project.

On this page:

- Introduction
- Enabling the Moderator Role
- Removing the Moderator from an Existing Project
- Adding the Moderator to an Existing Project

Introduction

By default, Crucible projects do not have a moderator. This allows for a streamlined review handling process, where the review author is the sole person who starts and stops the review. Projects in Crucible can have the moderator role enabled or disabled.

The setting for enabling moderators can only be set by a Crucible user with admin privileges.

Enabling the Moderator Role
The moderator role is configurable in Crucible as a per-project setting. By default, all reviews have an author and a moderator. However, the moderator role can be disabled.

To enable or disable the moderator role on a project:

1. Open the Crucible Admin screen, by clicking the 'Administration' link at the bottom of any Crucible page.
2. The Crucible Admin screen opens. Click 'Project List' under 'Project Settings' in the left navigation bar.
3. The Projects List screen opens. Click the 'Edit' link in the 'Edit Crucible Settings' column.
4. The 'Edit Project' screen opens.

**Screenshot: Disabling the Moderator in Crucible**

**Screenshot: Moderator Status Notification in the Projects List**

Removing the Moderator from an Existing Project

When you remove the moderator role from an existing project, be aware of the following points:

- Reviews created after the setting change will have the moderator role removed.
- Reviews created prior to the change will still retain the moderator they were assigned.
- If the removal of the moderator conflicts with other Crucible project settings, a warning will be shown on the Project List page.

If in doubt about the impact of adding or removing the moderator, you can create a new project (and set the moderator status during the project's creation).

Adding the Moderator to an Existing Project

If you add the moderator role back in to an existing project, be aware of the following points:

- Reviews created after the setting change will have the moderator role added.
- Reviews created prior to the change will still have no moderator.
- If the addition of the moderator conflicts with other Crucible project settings, a warning will be shown on the Project List page.
If in doubt about the impact of adding the moderator, you can create a new project (and set the moderator status during the project's creation).

**Setting Default Review Objectives**

To set default review objectives for all the reviews in a given project, carry out the following steps:

1. Open the 'Administration' page, 'Project List', then click 'Edit' to open the Edit Project screen.
2. In the 'Default Review Objectives Section', you can define some text that will appear by default in the 'Review Objectives' field of each new review. Click 'Done' to save your changes. This text can be edited, as any text contained in the Review Objectives text box can.

*Screenshot: Default Review Objectives in Crucible*

**Setting the Default Review Duration for a Project**

You can set a default time period (duration) that all reviews under a given project will run for. Reviews that are overdue will show up in red on the reviewer's dashboards.

To set a default duration for all reviews in a project:

1. From the 'Admin Menu', click 'Project List'.
2. The list of projects will be displayed. Click the 'Edit' link for the desired project, which appears to the right of the existing project name.
3. The 'Edit Project' page will be displayed. You can now adjust any of the given settings as desired.
4. In the 'Review Duration' section, you can define the default length of time (in week days) for reviews in this project. If you leave the field blank, then no time restriction is applied.

*Note that the 'Review Duration' only affects the default due date that appears when creating a review. The review's creator or moderator can specify a different date if they wish.*

To see instructions for the other items on this page, see the documentation for Editing a Project.

**Configuring Repositories**

The instructions for configuring your repositories are different, depending on your Crucible setup:

- **If you are running Crucible with Atlassian's FishEye**, you can manage your repositories using FishEye. See the FishEye documentation for more information.

- **If you are running Crucible alone**, you still can manage your repositories using native repository access (bundled with Crucible). See the FishEye documentation for more information. Please also see the What happens if I decide to stop using FishEye with Crucible page for important information about light FishEye.

**Enabling Reviews from the Server File System in Crucible**

To set up the File System as a Code Repository in stand-alone Crucible:

1. Start Crucible then open the 'Admin' menu.
2. Under the 'System Settings' heading, click 'Plugins' in the left-hand navigation bar.
3. The 'Plugins' screen opens.
4. Next to 'crucible-filesystem-scm-plugin', click 'Enable'.
5. New options appear next to 'File System SCM': 'Disable' and 'Configure'. Click 'Configure'.
6. The 'Configure Plugin' screen opens. Click 'Add Repository'.
7. The 'Add Repository' screen opens. Fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>What to enter</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Choose a unique name for the repository.</td>
</tr>
<tr>
<td>Base Path</td>
<td>Choose the lowest level of directory that Crucible will access.</td>
</tr>
</tbody>
</table>

8. Click 'Save'. The view will return to the list of repositories.
9. The server file system is now set up as a code repository for Crucible. You will be able to select files from it when creating reviews, with the ability to browse the files and directories on the hard drive.

### Setting Up a Git Repository in Crucible

This page describes how to use Crucible's native repository access to connect to an external Git repository. This native access uses a FishEye component (without requiring a FishEye license) which is why the description below refers to FishEye in places. See [What happens if I decide to stop using FishEye with Crucible](#) for more information.

Crucible interacts with Git repositories by executing the Git command in a separate process. Hence, the server running Crucible needs to have Git installed. Crucible indexes Git repositories by making a private, bare clone of your repository within Crucible's cache area. It uses this private clone for most Git operations.

ℹ️ The server running Crucible must have Git (version 1.7.1.1 or later) installed.

### Git Repository Details

#### Step 1

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Repository Type</td>
<td>Select 'Git'.</td>
</tr>
<tr>
<td>Name</td>
<td>Enter a name for this repository. The name may contain alphanumeric, underscore, '-' or '.' characters. This will be repository name in FishEye.</td>
</tr>
<tr>
<td>Description</td>
<td>Enter a short description of this repository.</td>
</tr>
</tbody>
</table>

#### Step 2

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Documentation for Crucible 2.8

**Step 3**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Store Diff Info</td>
<td>Tick this checkbox, if you want FishEye to cache information about file diffs in its database. This is required for some FishEye features. See Configuring Repository Details for more information on this setting.</td>
</tr>
<tr>
<td>Enable Repository After Adding</td>
<td>Tick this checkbox, to enable the repository after adding (i.e. when you click the 'Add' button).</td>
</tr>
</tbody>
</table>

**Limitations**

When FishEye indexes a Git repository, it indexes by the available branches. As it processes the commits on a branch, FishEye will assign the commit to the branch it first sees the commit on. Commits are only indexed once.

---

**Repository Location**

Enter the URL describing the Git repository location. FishEye will clone this repository for indexing purposes. You can use any URL recognised by Git itself. FishEye recognizes the following URL's:

- `git://server_name[:port]/path_to_repository`
- `http://server_name[:port]/path_to_repository`
- `https://server_name[:port]/path_to_repository`
- `ssh://server_name[:port]/path_to_repository`
- `file:///[hostname]/path_to_repository`

Do not use spaces in your URL.

FishEye does not directly support the ssh protocol for Git repositories. To use ssh, launch FishEye within the context of ssh-agent, to which the appropriate ssh keys have been added.

**Path**

(optional) Enter the path within the Git repository that you want FishEye to index. This lets you limit FishEye to indexing a subset of the complete Git repository.

**Block Size**

(optional) Enter how many commits you want FishEye to process in one batch. Larger values require more memory and increase the amount of work FishEye commits to the database in a single operation.

**Command Timeout**

(optional) Enter the time that a single Git command is allowed to take to execute. Any command that exceeds this time is terminated and the operation will fail.

**Rename Detection**

(optional) Select which Git rename detection strategy FishEye will use to detect copy and move operations within the repository. Please refer to the Git documentation for more information.
so if a commit belongs to multiple branches, the commit will not indexed against subsequent branches.

**Setting Up a Perforce Repository in Crucible Alone**

This page contains instructions on how to configure the Crucible Perforce SCM plugin to access Perforce repositories.

**Crucible SCM plugins superseded by Native Repository Access**

Crucible now ships with native repository access, which allows you to connect to repositories without a working version of FishEye. Crucible SCM plugins will still work, but we recommend that you stop using them in favour of native repository access. See What happens if I decide to stop using FishEye with Crucible for instructions.

On this page:

- Setting Up a Perforce Repository in Crucible Alone
- Notes

**Setting Up a Perforce Repository in Crucible Alone**

**To set up Perforce in Crucible alone:**

1. Ensure that the Perforce executable file is on the system path, in the Crucible server's Environment Variables.
2. Start Crucible then open the 'Admin' menu by clicking the Administration link in the footer of the page.
3. Under the 'Repository Settings' heading, click 'Repository List' in the left-hand navigation bar.
4. The 'Repository List' screen opens.
5. Find the Perforce repository plugin and click its Configure Plugin link.
6. The 'Configure Plugin' screen opens. Click 'Add Repository'.
7. The 'Add Repository' screen opens. Fill in the fields.

<table>
<thead>
<tr>
<th>Name</th>
<th>Choose a unique name for the repository.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Repository Server</td>
<td>Enter the base URL and port for the repository, for example: example.com:666.</td>
</tr>
<tr>
<td>Repository Path</td>
<td>Add the path to your Perforce repository. For example: //depot/code/example/main.</td>
</tr>
<tr>
<td>Perforce Username</td>
<td>Enter the username of the Perforce account that Crucible will use. (optional)</td>
</tr>
<tr>
<td>Perforce Password</td>
<td>Enter the password of the Perforce account that Crucible will use. (optional)</td>
</tr>
</tbody>
</table>

8. Click 'Save'. The view will return to the list of repositories.
9. Your Perforce repository is now set up for Crucible. You will be able to select changesets from it when creating reviews.

**Notes**
There is no ‘initial scanning’ required in this process, as Crucible's access to Perforce (when running alone) is strictly on-demand. Data is not indexed, hence there is no scanning.

Setting Up a Repository via FishEye

To use FishEye to access an external source control repository, such as Subversion or Git, for Crucible, see the FishEye documentation for how to add a repository.

- CVS
- Git
- Mercurial
- Perforce
- Subversion

Building index and cache

FishEye needs to build an index and cache of the contents of your repository, so some information will not appear in FishEye until this is complete. This may take some time to complete, depending on the size of the repositories.

We recommend you access the repository with a user that has only read access to the repository.

Setting Up a Subversion Repository in Crucible Alone

This page contains instructions on how to configure the Crucible Subversion SCM plugin to access Subversion repositories.

Crucible SCM plugins superseded by Native Repository Access

Crucible now ships with native repository access, which allows you to connect to repositories without a working version of FishEye. Crucible SCM plugins will still work, but we recommend that you stop using them in favour of native repository access. See What happens if I decide to stop using FishEye with Crucible for instructions.

On this page:

- Setting Up a Subversion Repository in Crucible Alone
  - Finding your Repository Root.

Setting Up a Subversion Repository in Crucible Alone

To set up Subversion in Crucible alone:

1. Start Crucible then open the 'Admin' menu by clicking the Administration link in the footer of the page.
2. Under the 'Repository Settings' heading, click 'Repository List' in the left-hand navigation bar.
3. The 'Repository List' screen opens.
4. Find the SVN repository plugin and click its Configure Plugin link.
5. The 'Configure Plugin' screen opens. Click 'Add Repository'.
6. The 'Add Repository' screen opens. Fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>What to enter</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Choose a unique name for the repository.</td>
</tr>
</tbody>
</table>
Repository Root | Enter the repository root URL for the repository. If you are not sure what the repository root is, please see the instructions below under "Finding your Repository Root".

Repository Path | Add the path on the base URL where your repository. For example, if you used the root URL above, and the full path to your Subversion instance is 'http://svn.example.com/svn5/', you would enter 'svn5' into this field.

SVN Username | Enter the username of the Subversion account that Crucible will use. 

SVN Password | Enter the password of the Subversion account that Crucible will use.

7. Click 'Save'. The view will return to the list of repositories.
8. Your Subversion repository is now set up for Crucible. You will be able to select changesets from it when creating reviews.

There is no 'initial scanning' required in this process, as Crucible's access to Subversion (when running alone) is strictly on-demand. Data is not cached, hence scanning is not required.

**Finding your Repository Root.**

Run the following command:

```
svn info SVN_URL
```

Where SVN_URL is the complete URL of the repository you want to add.

You will get something like the following:

```
>svn info http://svn.example.com/svn5/
Path: svn5
URL: http://svn.example.com/svn5/
Repository Root: http://svn.example.com/
Repository UUID: ce062a09-193b-427a-a7b3-a85007076e5d
Revision: 83
Node Kind: directory
Last Changed Author: ryan
Last Changed Rev: 83
Last Changed Date: 2009-05-07 10:48:41 +1000 (Thu, 07 May 2009)
```

Next to "Repository Root" is the URL you should define as your repository root. The path will be whatever is remaining.
Customising the Welcome Message

To customise the welcome message which is shown when Crucible opens, access the administration page and click ‘Customize Front Page’ under ‘Global Settings’ on the left navigation bar.

The ‘Customize Front Page Messages’ page opens.

On this page, you can provide your own custom text for the Crucible welcome message that is displayed to users when they first log in. You can also provide custom Support text, providing the contact details of your own support organisation, which also appears on the opening page.

You can enter text into the boxes provided for either message and click the small ‘Save Welcome Message’ or ‘Save Support Message’ button to save it, or enter text for both messages and click ‘Save All’. The changes are made immediately.

Screenshot: Crucible Customize Welcome and Support Messages

Using HTML

The content in the welcome screen can be arranged using basic HTML tables, image references or anchor tags such as the following:

```html
<a href="http://www.atlassian.com">Link to Atlassian Home Page</a>
```

Restoring the default messages

To revert to the default Welcome or Support messages, simply delete all text shown in the text box and click the corresponding ‘Save’ button.

Manually editing the opening screen

You can also directly edit the XML file that contains the welcome and support messages. This file is called config.xml, located in your installation folder.
To do this, simply add the following XML tags to `config.xml`:

```xml
<content>
  <front-page-message>Example welcome message here</front-page-message>
  <support-message>Example support message here</support-message>
</content>
```

### Customising email notifications

Email notifications in Crucible can be customised to change their formatting, by editing template files. This page contains instructions for this process.

#### Editing Crucible email templates

Template files for Crucible are stored in the `<FishEye home directory>/templates/` folder.

For Crucible, the set of templates is for plain-text email only. Note that these templates do not support embedding full diffs into notifications. They are only for changing the appearance and order of certain content inside the messages.

If you edit the templates of an operational Crucible instance, you may disrupt notifications that are being sent at that time. To avoid this, shut Crucible down during template editing.

#### Editing the subject line

1. Open the `crucible-notification-subject.ftl` template file from `<FishEye home directory>/templates/` in a text editor.
2. Type in your new text for the email subject, ensuring that all of the content is contained within line 1 of the template. `crucible-notification-subject.ftl` is used as the subject template for all Crucible email notifications.
3. Save and close the file.
4. Restarting Crucible will activate the new templates.

#### Editing the header

Header information will be included at the beginning of the email body text.

1. Open the `crucible-notification-header.ftl` template file from `<FishEye home directory>/templates/` in a text editor.
2. Add your new header content. `crucible-notification-header.ftl` is used as the header template for all Crucible email notifications.
3. Save and close the file.
4. Restarting Crucible will activate the new templates.

#### Editing the footer

Footer information will be included at the end of the email body text.

1. Open the `crucible-notification-footer.ftl` template file from `<FishEye home directory>/templates/` in a text editor. `crucible-notification-footer.ftl` is used as the footer template for all Crucible email notifications.
2. Add your new footer content.
3. Save and close the file.
4. Restarting Crucible will activate the new templates.
After an edit, the change to the email template will take place immediately. No restart is required.

Try and avoid editing the live template file, as Crucible may try to use it while you are editing. This could have unpredictable results. Instead, back up the template file (it's wise to keep original versions of all these files), edit a copy you have made, then overwrite the 'live' template once you have finished.

**Advanced editing of Crucible email templates**

The email notification templates use the [Freemarker](https://freemarker.org) format. Freemarker is a general templating engine enabling automated content.

If you are familiar with Freemarker, more advanced customisations can be made to the email notification templates. However, you make such adjustments at your own risk.

Note: In Crucible, email notifications are limited to plain-text format only.

**Crucible email template files**

The following template files for Crucible notification are stored in the `<FishEye home directory>/templates/` folder.

<table>
<thead>
<tr>
<th>Template filename</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>crucible-notification-subject.ftl</td>
<td>Subject template</td>
</tr>
<tr>
<td>crucible-notification-header.ftl</td>
<td>Header template</td>
</tr>
<tr>
<td>crucible-notification-footer.ftl</td>
<td>Footer template</td>
</tr>
<tr>
<td>state-closed-notification.ftl</td>
<td>State Closed template</td>
</tr>
<tr>
<td>all-completed-notification.ftl</td>
<td>All Completed template</td>
</tr>
<tr>
<td>state-changed-notification.ftl</td>
<td>State Changed template</td>
</tr>
<tr>
<td>completed-notification.ftl</td>
<td>Completed template</td>
</tr>
<tr>
<td>general-notification.ftl</td>
<td>General notification template</td>
</tr>
<tr>
<td>uncompleted-notification.ftl</td>
<td>'Uncompleted' template</td>
</tr>
<tr>
<td>all-no-longer-completed-notification.ftl</td>
<td>All-No-Longer-Completed template</td>
</tr>
<tr>
<td>comment-notification.ftl</td>
<td>Comment template</td>
</tr>
<tr>
<td>reply-notification.ftl</td>
<td>Reply template</td>
</tr>
<tr>
<td>review-precis-plain.ftl</td>
<td>Precis template</td>
</tr>
<tr>
<td>review-reminder-notification.ftl</td>
<td>Review Reminder template</td>
</tr>
</tbody>
</table>

See also [Customising FishEye Email Notifications](#).
Freemarker Data Model for Email Templates

Customising Crucible email templates with Freemarker

See the Freemarker documentation for instructions on Freemarker syntax. Use the templates that ship with Crucible as a guide to the properties available on each object.

Specific email types will have extra data associated with them, and this data will be available in that particular template (but not in others).

Example

The syntax to access the data-model, using the data model object ‘link’ as an example, place this code into the email at the desired position.

```text
${notification.link}
```

Setting up Users and Security

User management and security settings are covered in the FishEye documentation.

Other Related Security Resources

- How to Report a Security Issue
- Security Advisory Publishing Policy
- Security Patch Policy
- Severity Levels for Security Issues
- Configuring User Managed Mappings
- Setting up Users and Security
- Creating a Permission Scheme

Customising the Defect Classifications

This page explains how to customise defects and their classifications in Crucible.

On this page:

- Defects in Crucible Comments
- Changing Classification Settings
- Default Crucible Classifications
  - Ranking
  - Classification

Defects in Crucible Comments

Defects are comments made by reviewers that indicate a problem in a review. Defects can be classified by rank and type, custom classifications can also be defined. The default classifications are shown in the screenshot below.

Screenshot: The List of Defect Classifications
Changing Classification Settings

To change the default classifications:

1. Open the Crucible Admin screen. The ‘Admin Menu’ opens.
2. Click ‘Customize Crucible Defect Classifications’ under ‘Global Settings’ in the Admin Menu.

*Only Crucible Admin users have access to this screen.*

Any changes made within ‘Customize crucible defect classifications’ will only affect reviews created after the setting is changed.

Default Crucible Classifications

There are two default defect classifications that are preset in Crucible; ranking and classification. These settings (and their sub-categories) can be edited or removed; other custom classifications can be added.

Ranking

This classification can be set to 'Major' or 'Minor', indicating the importance of the defect.

Classification

This setting helps to define the nature of the defect in particular detail. This classification can be set to one of the options in the following table; the meaning of these is detailed in the table below.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Missing</td>
<td>The defect applies to code or information that is missing (absent).</td>
</tr>
<tr>
<td>Extra (superfluous)</td>
<td>The defect applies to code or information that should be removed.</td>
</tr>
<tr>
<td>Classification</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Ambiguous</td>
<td>The defect applies to code or information that is not clear or easy to understand.</td>
</tr>
<tr>
<td>Inconsistent</td>
<td>The defect applies to code or information that is applied in several different ways.</td>
</tr>
<tr>
<td>Improvement desirable</td>
<td>The defect applies to code or information that needs to be revised.</td>
</tr>
<tr>
<td>Not conforming to standards</td>
<td>The defect applies to code or information that breaks established conventions.</td>
</tr>
<tr>
<td>Risk-prone</td>
<td>The defect applies to code or information that takes unacceptable risks.</td>
</tr>
<tr>
<td>Factually incorrect</td>
<td>The defect applies to code or information that is wrong.</td>
</tr>
<tr>
<td>Not implementable</td>
<td>The defect applies to code or information that may be impossible to create.</td>
</tr>
<tr>
<td>Editorial</td>
<td>The defect applies to code or information where the classification as a defect may be subject to personal opinion.</td>
</tr>
</tbody>
</table>

Screenshot: Editing Defect Classifications in Crucible
Creating a Permission Scheme

This page contains information on how to create a permission scheme in Crucible.

On this page:

- Introduction to Crucible Permissions
- Creating a Permission Scheme
- Editing a Permission Scheme
- List of Crucible Permissions
- Further Reading

Introduction to Crucible Permissions

The page permission does not exist.

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.

Creating a Permission Scheme

To create a permission scheme:

1. From the 'Admin Menu', click 'Permission Schemes'.
2. The 'Permission Schemes' page will be displayed, showing a list of existing permission schemes. Click the 'Create a New Permission Scheme' link, which appears below the list.
3. In the 'Name' field, type a short phrase to uniquely identify your project (see screenshot 1 below).
4. Click the 'Create' button to create your new permission scheme. The 'Edit Permission Scheme' page will be displayed for your new permission scheme (see screenshot two, below).
   Your new permission scheme will have the default assignees shown in the table above.
5. To edit the assignees for a permission, click the 'Edit' link corresponding to the permission. The 'Edit Permission Scheme' page will be displayed.
6. Choose the appropriate assignee(s) for this permission:
   • To assign this permission to anonymous users, select the 'Allow Anonymous users' check-box.
   • To assign this permission to all logged-in users, select the 'Allow All logged in users' check-box.
   • To assign this permission to a particular user, type their username into the 'Individual users' field (hint: you can type just part of the name, then press <Enter> to select from a list of matching usernames).
   • To assign this permission to a particular group of users, type the group name into the 'Groups' field (hint: you can type just part of the group name, then press <Enter> to select from a list of matching groups).
   • To assign this permission to users who belong to a particular participant('Reviewer' / 'Moderator' / 'Author' / 'Creator'), select the corresponding check-box.
7. Click the 'Save' button.

Next step: see Associating a Permission Scheme with a Project.

Screenshot 1: Adding a Permission Scheme

<table>
<thead>
<tr>
<th>Name</th>
<th>Projects using this scheme</th>
<th>create</th>
<th>cancel</th>
</tr>
</thead>
<tbody>
<tr>
<td>Top Secret</td>
<td>Default Project</td>
<td>edit</td>
<td>copy</td>
</tr>
</tbody>
</table>

Editing a Permission Scheme

To edit a permission scheme:

1. From the 'Admin Menu', click 'Permission Schemes'.

Created by Atlassian in 2012. Licensed under a Creative Commons Attribution 2.5 Australia License.
2. Click ‘edit’ next to the scheme you wish to change. The ‘Edit Permission Scheme’ page will be displayed.
3. On the ‘Edit Permission Scheme’ page, you can change the groups or users that are allowed individual permissions by clicking ‘edit’ next to the permission in question.
4. When you have finished editing, click the ‘Save’ button.

**Screenshot: Edit a Permission Scheme**

<table>
<thead>
<tr>
<th>Permissions</th>
<th>Users / Groups / Review Roles</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit Review Details</td>
<td>- Anonymous users: false</td>
</tr>
<tr>
<td></td>
<td>- All logged in users: false</td>
</tr>
<tr>
<td></td>
<td>- Individual users:</td>
</tr>
<tr>
<td></td>
<td>- Groups:</td>
</tr>
<tr>
<td></td>
<td>- Review Roles: Moderator Creator</td>
</tr>
<tr>
<td>View</td>
<td>- Anonymous users: true</td>
</tr>
<tr>
<td></td>
<td>- All logged in users: true</td>
</tr>
<tr>
<td></td>
<td>- Individual users:</td>
</tr>
<tr>
<td></td>
<td>- Groups:</td>
</tr>
<tr>
<td></td>
<td>- Review Roles: Moderator Reviewer Author Creator</td>
</tr>
<tr>
<td>Abandon</td>
<td>- Anonymous users: false</td>
</tr>
<tr>
<td></td>
<td>- All logged in users: false</td>
</tr>
<tr>
<td></td>
<td>- Individual users:</td>
</tr>
<tr>
<td></td>
<td>- Groups:</td>
</tr>
<tr>
<td></td>
<td>- Review Roles: Moderator Creator</td>
</tr>
<tr>
<td>Re-Open</td>
<td>- Anonymous users: false</td>
</tr>
<tr>
<td></td>
<td>- All logged in users: false</td>
</tr>
<tr>
<td></td>
<td>- Individual users:</td>
</tr>
<tr>
<td></td>
<td>- Groups:</td>
</tr>
<tr>
<td></td>
<td>- Review Roles: Moderator Creator</td>
</tr>
<tr>
<td>Uncomplete</td>
<td>- Anonymous users: false</td>
</tr>
<tr>
<td></td>
<td>- All logged in users: false</td>
</tr>
<tr>
<td></td>
<td>- Individual users:</td>
</tr>
<tr>
<td></td>
<td>- Groups:</td>
</tr>
<tr>
<td></td>
<td>- Review Roles: Reviewer</td>
</tr>
<tr>
<td>Reject</td>
<td>- Anonymous users: false</td>
</tr>
<tr>
<td></td>
<td>- All logged in users: false</td>
</tr>
<tr>
<td></td>
<td>- Individual users:</td>
</tr>
<tr>
<td></td>
<td>- Groups:</td>
</tr>
</tbody>
</table>

**Screenshot: Editing the 'View' Permission**

**List of Crucible Permissions**

The following permissions are available:

<table>
<thead>
<tr>
<th>Permission</th>
<th>Description</th>
<th>Default Assignees</th>
</tr>
</thead>
</table>

*Created by Atlassian in 2012. Licensed under a Creative Commons Attribution 2.5 Australia License.*
<table>
<thead>
<tr>
<th>Action</th>
<th>Description</th>
<th>Permissions</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>'Summarise'</td>
<td>Ability to summarise a review. (Normally this would be done after all reviewers have completed their review.)</td>
<td>'Moderator'</td>
</tr>
<tr>
<td>'Create'</td>
<td>Ability to create a review.</td>
<td>All logged-in users</td>
</tr>
</tbody>
</table>
### Recover

Ability to resurrect an abandoned (i.e. cancelled) review.

- **Creator**
- **Moderator**

### Further Reading

For more information on permissions schemes in Crucible, see the following documentation pages:

- [Agile Permissions Schemes in Crucible](#)
- [Associating a Permission Scheme with a Project](#)

### Agile Permissions Schemes in Crucible

This page contains information about using and editing Agile permission schemes in Crucible.

#### Understanding the Agile Permissions Scheme

Agile development teams may not want to use the default Crucible permission schemes that require one person to approve or summarise reviews. Crucible ships with a pre-defined Agile permission scheme. By Agile, we mean permission schemes that have no moderator and very liberal permissions, suited to Agile or self-organising teams.

To use the Agile permissions scheme when creating a project, simply select ‘agile’ from the drop down list under ‘Project Permissions Scheme’ on the ‘Edit Project’ screen.

#### Considerations

- **If you began your installation of Crucible with Crucible 2.1 or later, then this permission scheme will appear in the list of permission schemes in the administration menu.**

- **If you have upgraded from an earlier version of Crucible (pre Crucible 2.0), then the Agile permission may not appear by default. However, you can still create the equivalent by disabling the moderator role when creating projects, allowing freer access to summarising, closing and generally tending to Crucible reviews.**

- **If you disable the moderator role on the Edit Project screen, then Crucible will check the current permission scheme. If the current permission scheme requires a moderator, a warning will be shown and you will be prompted to create a new permission scheme which will be called ‘Agile’ (or Agile-X if the name Agile already exists, where X is a number appended to the scheme name). The new permissions scheme will not require a moderator to carry out any actions.**

![Screenshot: Warnings on Permission Schemes that Require a Moderator](#)

#### Associating a Permission Scheme with a Project

### Agile Permissions Schemes in Crucible

This page contains information about using and editing Agile permission schemes in Crucible.

#### Understanding the Agile Permissions Scheme

Agile development teams may not want to use the default Crucible permission schemes that require one person to approve or summarise reviews. Crucible ships with a pre-defined Agile permission scheme. By Agile, we mean permission schemes that have no moderator and very liberal permissions, suited to Agile or self-organising teams.

To use the Agile permissions scheme when creating a project, simply select ‘agile’ from the drop down list under ‘Project Permissions Scheme’ on the ‘Edit Project’ screen.

#### Considerations

- **If you began your installation of Crucible with Crucible 2.1 or later, then this permission scheme will appear in the list of permission schemes in the administration menu.**

- **If you have upgraded from an earlier version of Crucible (pre Crucible 2.0), then the Agile permission may not appear by default. However, you can still create the equivalent by disabling the moderator role when creating projects, allowing freer access to summarising, closing and generally tending to Crucible reviews.**

- **If you disable the moderator role on the Edit Project screen, then Crucible will check the current permission scheme. If the current permission scheme requires a moderator, a warning will be shown and you will be prompted to create a new permission scheme which will be called ‘Agile’ (or Agile-X if the name Agile already exists, where X is a number appended to the scheme name). The new permissions scheme will not require a moderator to carry out any actions.**

![Screenshot: Warnings on Permission Schemes that Require a Moderator](#)

#### Associating a Permission Scheme with a Project
This page explains how to associate a permission scheme with a Crucible project and show details of the default permission schemes included with Crucible.

On this page:

- **Associating a Permission Scheme with a Crucible Project**
- **Overview of the Permission Schemes Bundled with Crucible**
  - Default Permission Scheme Settings
  - Agile Permission Scheme Settings
- **Related Links**

### Associating a Permission Scheme with a Crucible Project

To associate a permission scheme with a project:

1. From the 'Admin Menu', click 'Project List'.
2. The 'Projects List' page will be displayed. Find the project you wish to associate with your permission scheme, and click its 'Edit' link.
3. The 'Edit Project' page will be displayed.
4. Under the heading 'Project Permissions Scheme', click the 'Permission Scheme' drop-down list to select your permission scheme.
   - You will be shown a list of the schemes that have been created in Crucible. You can create a new permission scheme if necessary.
5. Click the 'Save' button.

### Overview of the Permission Schemes Bundled with Crucible

Crucible comes with two permission schemes. 'Default' and 'Agile'. The following tables show the default settings in detail; note that these can be easily edited by admin users to suit your needs.

#### Default Permission Scheme Settings

This table shows the various permissions and which user groups have them by default.

<table>
<thead>
<tr>
<th>Permission</th>
<th>Anonymous</th>
<th>All Logged In</th>
<th>Individuals</th>
<th>Groups</th>
<th>Review Roles</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abandon</td>
<td>false</td>
<td>false</td>
<td>None</td>
<td>None</td>
<td>Creator, Moderator</td>
</tr>
<tr>
<td>Approve</td>
<td>false</td>
<td>false</td>
<td>None</td>
<td>None</td>
<td>Moderator</td>
</tr>
<tr>
<td>Close</td>
<td>false</td>
<td>false</td>
<td>None</td>
<td>None</td>
<td>Moderator</td>
</tr>
<tr>
<td>Comment</td>
<td>false</td>
<td>false</td>
<td>None</td>
<td>None</td>
<td>Reviewer, Creator, Author, Moderator</td>
</tr>
<tr>
<td>Complete</td>
<td>false</td>
<td>false</td>
<td>None</td>
<td>None</td>
<td>Reviewer</td>
</tr>
<tr>
<td>Create</td>
<td>false</td>
<td>true</td>
<td>None</td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td>Delete</td>
<td>false</td>
<td>false</td>
<td>None</td>
<td>None</td>
<td>Creator, Moderator</td>
</tr>
</tbody>
</table>
# Agile Permission Scheme Settings

This table shows the various permissions and which user groups have them by default.

<table>
<thead>
<tr>
<th>Permission</th>
<th>Anonymous</th>
<th>All Logged In</th>
<th>Individuals</th>
<th>Groups</th>
<th>Review Roles</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abandon</td>
<td>false</td>
<td>false</td>
<td>None</td>
<td>None</td>
<td>Creator, Author, Moderator</td>
</tr>
<tr>
<td>Approve</td>
<td>false</td>
<td>false</td>
<td>None</td>
<td>None</td>
<td>Creator, Author, Moderator</td>
</tr>
<tr>
<td>Close</td>
<td>false</td>
<td>false</td>
<td>None</td>
<td>None</td>
<td>Reviewer, Creator, Author, Moderator</td>
</tr>
<tr>
<td>Comment</td>
<td>false</td>
<td>false</td>
<td>None</td>
<td>None</td>
<td>Reviewer, Creator, Author, Moderator</td>
</tr>
<tr>
<td>Complete</td>
<td>false</td>
<td>false</td>
<td>None</td>
<td>None</td>
<td>Reviewer</td>
</tr>
<tr>
<td>Create</td>
<td>false</td>
<td>true</td>
<td>None</td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td>Delete</td>
<td>false</td>
<td>false</td>
<td>None</td>
<td>None</td>
<td>Creator, Author, Moderator</td>
</tr>
<tr>
<td>Edit Review Details</td>
<td>false</td>
<td>false</td>
<td>None</td>
<td>None</td>
<td>Reviewer, Creator, Author, Moderator</td>
</tr>
</tbody>
</table>

The default permission scheme has changed since Crucible 1.6.
In Crucible, Administrators can control whether users can use the Author Mapping setting to map their own Crucible usernames to repository committer accounts or not. By default, the setting allows users to set their own mappings.

If you wish to lock down the mappings for security or audit reasons, this setting lets you restrict all management of mappings to Crucible administrators only.

To do this, click ‘Administration’ in the footer of the Crucible interface and then ‘Security’ in the left navigation bar. The ‘Authentication Settings’ page opens. You can click to set User Managed Mappings ‘On’ or ‘Off’. The setting is applied immediately.

Related Links
- Creating a Permission Scheme

Configuring User Managed Mappings

Steps
• Stop Fisheye/Crucible,
• and then create the file `FISHEYE_HOME/content/WEB-INF/jetty-web.xml` with the following content:

```xml
<?xml version="1.0"  encoding="ISO-8859-1"?>
<!DOCTYPE Configure PUBLIC "-//Mort Bay Consulting//DTD Configure//EN"
"http://www.eclipse.org/jetty/configure.dtd">

<Configure class="org.mortbay.jetty.webapp.WebAppContext">
  <Call name="addHandler">
    <Arg>
      <New class="com.cenqua.fisheye.web.jetty.FishEyeRequestLogHandler">
        <New id="RequestLogImpl" class="org.mortbay.jetty.NCSAResquestLog">
          <Arg><SystemProperty name="jetty.logs" default="../var/log"/>
        </New>
        <Set name="requestLog">
          <New id="RequestLogImpl" class="org.mortbay.jetty.NCSARequestLog">
            <Arg><SystemProperty name="jetty.logs" default="../var/log"/>
          </New>
          <Set name="retainDays">90</Set>
          <Set name="append">true</Set>
          <Set name="extended">false</Set>
          <Set name="LogTimeZone">GMT</Set>
        </Set>
      </New>
    </New>
  </Call>
</Configure>
```

• Restart Fisheye/Crucible.

This will create an access log in `FISHEYE_HOME/var/log/fisheye-access.log.yyyy_mm_dd` (e.g., `fisheye-access.log.2010_03_17`). If you want to change the path to your `FISHEYE_INST` directory, change the `default="../var/log"` to the path to the log folder in `FISHEYE_INST`.

⚠️ **The log directory must exist**

If the path to the log directory given by the `default` attribute of the `SystemProperty` tag (defined in the line 10 in the `jetty-web.xml` above) does not exist, then FishEye will fail to start and will not log any error message.

The path given in the example below is correct when `FISHEYE_INST` and `FISHEYE_HOME` are the same directory, otherwise please use the absolute path of your `FISHEYE_INST/var/log` directory.

**Log format**

The logs are written in [NCSA format](#).
Compatibility

If you are using an earlier version of FishEye than FishEye 2.7.8, replace com.cenqua.fisheye.web.jetty.
FishEyeRequestLogHandler by org.mortbay.jetty.handler.RequestLogHandler.

FishEyeRequestLogHandler was added in 2.7.8 to fix an issue where the user credentials would not be added to the NCSA log: FE-3040.

Backing up and restoring Crucible data

Crucible data can be backed up from the admin interface or command line. This page contains the command syntax, options and the required procedure to backup and restore your Crucible instance.

Backing up Crucible data

The Crucible admin backup process

1. Navigate to the Crucible 'Admin' screen (click the 'Administration' link in the footer of any Crucible page).
2. On the Admin screen, click 'Backup' under the 'System' heading in the left navigation bar. The Backup screen opens.
3. On the Backup screen, the 'File Path' field indicates where the backup file (in .zip format) will be stored. You can manually edit this path to change it. Under the heading 'Include', a list of check boxes is shown, with the following items:
On this page:

- Backing up Crucible data
  - The Crucible admin backup process
  - The Crucible command line backup process
  - Components of a Crucible backup
  - Backup command line options
  - Command line examples
  - Advanced backup command line settings
  - Known limitations

- Scheduling Crucible backups
- Restoring Crucible data
  - Restoring Crucible data from the command line
  - Command line restore options
  - Advanced command line restore settings

- Plugins and their configuration data
- SQL database
- Web templates
- Uploaded files and local copies of files under review.
- Repository and application caches.

Repository and application caches contain temporary data stored from repository scans and library caches that improve startup time. Both will be recreated automatically by re-scanning the source repositories, so the backup files can be reduced by a significant amount by excluding these (if the cost of re-scanning is acceptable).

4. Once you have chosen your options, click 'Create Backup Now'.

Screenshot: The Crucible Backup Screen

The Crucible command line backup process

- Your Crucible instance must be running during the backup.
  1. Open a command line interface on the Crucible server computer.
  2. Navigate to the `<FishEye home directory>/bin/` directory.
  3. Run the backup command on the command line with the desired options.
  4. The backup is created as a new Zip archive file and placed in the `FISHEYE_INST/backup/` directory.
Components of a Crucible backup

The Crucible backup is highly configurable and allows for many different configurations. This table shows the various components of the backup, what they are for and how they can be used.

<table>
<thead>
<tr>
<th>Component</th>
<th>Purpose</th>
<th>Defaults</th>
</tr>
</thead>
<tbody>
<tr>
<td>SQL Database</td>
<td>Refers to the SQL content database (used by both FishEye and Crucible and containing all user profile data, reviews and their comments).</td>
<td>Backed up by default.</td>
</tr>
<tr>
<td>Cache</td>
<td>The cache contains data that reflects the state of FishEye’s repositories. Without it, FishEye must re-scan its repositories after a backup is restored. The cache also contains OSGI library data that increases startup time. These too can be excluded and will be generated automatically when the application is started.</td>
<td>The cache is not backed up by default as it tends to be large (running a risk of pushing the maximum file size for Java backups), whilst also representing replaceable data.</td>
</tr>
<tr>
<td>Plugins</td>
<td>Plugins are 3rd-party extensions that you may have installed, and configuration for all plugins (this includes configuration for Crucible's set of standard plugins).</td>
<td>Configuration data for all plugins are backed up by default, as well as all plugins installed in FISHEYE_INST/var/plugins/user.</td>
</tr>
<tr>
<td>Templates</td>
<td>In this context, these are custom freemarker templates that you or your users have created. They live in FISHEYE_INST/template.</td>
<td>Templates are backed up by default. You can choose to exclude them from the backup if your templates directory is covered by some other backup mechanism.</td>
</tr>
<tr>
<td>Uploads</td>
<td>In this context, uploads refers to files which are added to Crucible via the web interface (such as patch file reviews). It also includes each repository-backed file that went under review, when Crucible is configured to make a local copy of every reviewed file.</td>
<td>Uploads are backed up by default. You can choose not to back them up for example when the FISHEYE_INST/var/data/uploads directory is already covered by some other backup mechanism.</td>
</tr>
<tr>
<td>ActiveObjects</td>
<td>Configuration data stored by plugins</td>
<td>Backed up by default</td>
</tr>
</tbody>
</table>

Note that the backup will always include the configuration data (config.xml), your license file and the FishEye user data.
Backup command line options

These examples are for use in a Linux-like operating system. When using these commands on Windows, use the file name fisheyectl.bat and use the correct slashes. Run the command from the `<FishEye home directory>/bin/` directory.

The basic syntax of the backup command is as follows:

```
$ ./fisheyectl.sh backup [OPTIONS]
```

To see inline help for all backup options, run the following command in the `<FishEye home directory>/bin/` directory:

```
$ ./fisheyectl.sh backup --help
```

<table>
<thead>
<tr>
<th>Option</th>
<th>Switch</th>
<th>Description</th>
<th>Default setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quiet mode</td>
<td>-q OR --quiet</td>
<td>Suppresses output</td>
<td>No</td>
</tr>
<tr>
<td>Output filename</td>
<td>-f OR --file</td>
<td>Specify a different path and filename to the FISHEYE_INST/backup/backup_YYYY-DD-MM_HH_mm.zip file. When filename is omitted, the backup filename contains the date and time.</td>
<td>FISHEYE_INST/backup/ is the default directory.</td>
</tr>
<tr>
<td>Compression level</td>
<td>--compression OR -c</td>
<td>Sets the Zip compression level, from 1-9. Runs at level 6 if no argument is passed.</td>
<td>Yes (6)</td>
</tr>
<tr>
<td>Anonymise</td>
<td>-a OR --anonymise</td>
<td>Anonymises the SQL database by replacing all text with 'x'. This is only useful when sending a backup to Atlassian as part of a support case. Please do not anonymise data unless the Support Engineer handling your support case has specifically requested the data anonymised (as often anonymised data will not help reproduce the issue).</td>
<td>No</td>
</tr>
</tbody>
</table>
Cache Backup

Include the repository caching files in the backup. These hold information gained from scanning the repositories and can be quite large (many gigabytes). However, it can shorten the time needed to re-scan the repositories after data is restored.

No. By default, the cache data is excluded from backups.

Command line examples

These examples are for use in a Linux-like operating system. When using these commands on Windows, use the filename fisheyectl.bat and use the correct slashes. Run the command from the <FishEye home directory>/bin/directory.

Back up with compression of 9, quiet mode and setting an output location

```bash
$ ./fisheyectl.sh backup --compression 9 -q -f /application_backups/fisheye/20090215.zip
```

Backup including cache data (also includes all default components)

```bash
$ ./fisheyectl.sh backup --cache
```

Restoring a backup with cache data (also restores all default components)

```bash
$ ./fisheyectl.sh restore --cache
```

Advanced backup command line settings

In some cases it might be preferable to only backup a limited set of items. This could be useful when your instance uses an external database such as MySQL or PostgreSQL and your DBA has already configured automatic backups in the database. The commands below allow this.

<table>
<thead>
<tr>
<th>Option</th>
<th>Switch</th>
<th>Description</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exclude Plugins</td>
<td>--no-plugins</td>
<td>Excludes plugins from the backup.</td>
<td>No. By default, plugins are included in every backup.</td>
</tr>
<tr>
<td>Exclude Templates</td>
<td>--no-templates</td>
<td>Excludes templates from the backup.</td>
<td>No. By default, templates are included in every backup.</td>
</tr>
</tbody>
</table>
Exclude Uploads | --no-uploads | Excludes uploaded files (such as patch reviews, stored in Crucible's internal database) from the backup. | No. By default, uploads are included in every backup.

Exclude SQL Database | --no-sql | Excludes the SQL content database used by both FishEye and Crucible. | No. By default, this data is included in every backup.

Show help | --help OR -h | Shows inline help on the command line. | No

Known limitations

Please note that the below limitations are common for any Java based backup tool.

**Archives Containing Over 65535 Files**

Versions of Java earlier than v1.6 (b25) are incapable of handling zip files that contain more than 65,535 files. The solution for this problem is to either upgrade to a version of Java later than v1.6 (b25), or ensure that the archive does not exceed the threshold (contains less than 65,535 files). The FishEye cache (not included in backups by default) can be a contributor of many small files. Hence, exclude the cache from backups if this is likely to be a concern.

**Archives Larger Than 4GB**

Java has trouble reading and writing zip files that are larger than 4GB. As of release 1.5 Java appears capable of reliably creating archives that are over 4GB, but remains unable to extract them. For details see Sun's bug report. Also be aware of the fact that some file systems (including FAT32) have trouble with files larger than 4GB.

As a workaround, make sure you do not create archives that are larger than 4GB. The FishEye cache (not included in backups by default) can be a contributor of a lot of small files (although these tend to compress very well). If you still want to archive everything and end up with an archive that is too large, consider creating separate backups for the FishEye cache and uploaded files respectively.

**Scheduling Crucible backups**

To set a schedule for automatic backups, open the administration screen and click 'Backup' under 'System' on the left navigation bar. The 'Backup' page opens. Now, click the link 'Manage Scheduled Backups' at the bottom of the page. The 'Scheduled Backups' page opens.

On the 'Scheduled Backups' page, click 'Edit' to adjust the backup schedule. Set the desired options and click 'Save'.

The options for scheduled backups are detailed in the table below.

<table>
<thead>
<tr>
<th>Option name</th>
<th>Description</th>
<th>Allowed Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disable Scheduled Backups</td>
<td>Stops regular backups from taking place.</td>
<td>On (disabled) or Off (enabled)</td>
</tr>
<tr>
<td>Backup path</td>
<td>The path where the backup .zip file will be stored.</td>
<td>Any system or network path that FishEye or Crucible can access.</td>
</tr>
</tbody>
</table>
### Backup file prefix
Characters that will be added to the beginning of the backup file name.

### Backup file date pattern
Sets a date for the next (or initial) backup to take place.

### Backup frequency
Sets how often the backup will take place.

### Backup time (HH:mm)
The time when the backup will take place.

### Include
Specifies which items must be included in the backups (these components are explained at the top of this page).

---

**Screenshot: Scheduling Backups in FishEye and Crucible**

Be aware that scheduled backups can fill up disks unless you regularly move or delete old archives.

**Restoring Crucible data**

**Restoring Crucible data from the command line**

There is currently no way to restore a backup from the web interface because Crucible must be shut down.
during a data restore.

⚠️ Restoring a backup will irreversibly overwrite the data of your installation with the data from the backup archive.

If you made a backup from production which connected to an external database, and restore this backup to a test server without specifying another database to restore too, you will drop and restore to your production database. Thus when restoring to a test server, always ensure you specify the correct database to restore to (or restore to an in-built database).

1. Install Crucible into a new, empty directory (this must be the same version that the backup was created from, or later).
   ⚠️ Note that you cannot restore data into versions of Crucible which are older than the version that created the backup.

2. Make sure the Crucible instance is not running.
3. Open a command line interface on the Crucible server computer.
4. Run the restore command on the command line with the desired options.
5. The specified elements will be restored.
6. Start the Crucible instance.
7. When using FishEye integrated with Crucible, you will need to re-index your repositories after restoring data, unless the backup archive was created with the `--cache` option.

Command line restore options

ℹ️ These examples are for use in a Linux-like operating system. When using these commands on Windows, use the filename `fisheyectl.bat` and use the correct slashes. Run the command from the `<FishEye home directory>/bin/` directory.

The basic syntax of the restore command is as follows:

```bash
$ ./fisheyectl.sh restore -f /path/to/backup_2009-10-02_1138.zip [OPTIONS]
```

To see inline help for all backup options, run the following command in the `<FishEye home directory>/bin/` directory:

```bash
$ ./fisheyectl.sh restore --help
```

Restores a FishEye/Crucible backup instance.

If you are using an external database (as opposed to the default built-in database), make sure the JDBC driver file is present in the `FISHEYE_INST/lib` directory when running restore.

<table>
<thead>
<tr>
<th>Option</th>
<th>Switch</th>
<th>Description</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>Suppress output</td>
<td><code>--quiet OR -q</code></td>
<td>Suppress the output messages from the restore program on the command line.</td>
<td>No</td>
</tr>
<tr>
<td>Choose file to restore from</td>
<td><code>--file PATH/Filename</code> OR <code>-f PATH/Filename</code></td>
<td>Restore the backup from PATH/Filename.</td>
<td>Yes (required)</td>
</tr>
</tbody>
</table>
Show inline help | --help OR -h | Displays help for options on the command line. | No

**Advanced command line restore settings**

By default, the restore program will restore all items found in the backup archive (so if you included the caches using the `--cache` option, these will automatically be restored). However, it is possible to only restore a subset of items from the backup, by explicitly specifying the item names on the command line and only those will be restored.

<table>
<thead>
<tr>
<th>Option</th>
<th>Switch</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Restore FishEye cache</td>
<td>--cache</td>
<td>Restore the repository cache backup.</td>
</tr>
<tr>
<td>Restore plugins</td>
<td>--plugins</td>
<td>Restore 3rd-party plugins and their configuration data.</td>
</tr>
<tr>
<td>Restore templates</td>
<td>--templates</td>
<td>Restore freemarker templates from the backup (the restored instance will use the built-in templates).</td>
</tr>
<tr>
<td>Restore uploads</td>
<td>--uploads</td>
<td>Restore uploads (e.g. patch files uploaded into Crucible and contents of files under review).</td>
</tr>
<tr>
<td>Restore Crucible reviews</td>
<td>--sql</td>
<td>Restore the SQL database containing user profiles, reviews and review comments.</td>
</tr>
<tr>
<td>Set database type</td>
<td>--dbtype OR -t</td>
<td>SQL database type (‘mysql’, ‘postgresql’ or ‘built-in’). Only required when restoring to a database location different to that used at backup time.</td>
</tr>
<tr>
<td>Set JDBC URL</td>
<td>--jdbcurl OR -j</td>
<td>JDBC URL of the SQL database. Only required when restoring to a database location different to that used at backup time (not applicable for ‘built-in’).</td>
</tr>
<tr>
<td>Set JDBC username</td>
<td>--username OR -u</td>
<td>JDBC username of the SQL database. Only required when restoring to a database location different to that used at backup time (not applicable for ‘built-in’).</td>
</tr>
</tbody>
</table>
**Documentation for Crucible 2.8**

---

### JDBC password

---password OR -p

| JDBC password of the SQL database. Only required when restoring to a database location different to that used at backup time (not applicable for 'built-in'). |

### JDBC class

---driver OR -d

| Specifies the JDBC driver class name needed to access the SQL database. Only required when restoring to a database location different to that used at backup time and when using a different JDBC driver than the standard driver associated with the database specified through --dbtype. (Not applicable for 'built-in'.) |

---

### Notes on migrating backup data

When the process restores a SQL database, it looks at the configuration data (config.xml) included in the backup archive to learn which database product was used and how to connect to it. When Crucible uses the built-in HSQLDB database (which is the default), the restored instance will also use that. However, when the restored instance will use a different database than the backed up instance (for instance, HSQLDB was used at the time the backup was created, but it needs to be restored on MySQL), use the command line options to point the process to the new database.

**Command line example: migrating backup data to MySQL**

These examples are for use in a Linux-like operating system. When using these commands on Windows, use the filename fisheyectl.bat and use the correct slashes. Run the command from the FishEye home directory.

Restoring to a Crucible instance that uses a different database (ensure the mysql driver jar file is present in the FISHEYE_INST/lib directory)

```
$ ./fisheyectl.sh restore \
   --username john \ 
   --password smith \ 
   --jdbcurl jdbc:mysql://localhost:3306/crucible \ 
   --dbtype mysql \ 
   --file /path/to/backup_2009-10-02_1138.zip
```

---

### Best Practices for Crucible Configuration

1. **Set up a separate FISHEYE_INST folder location on the same system for Crucible’s data.**

   This will allow for easy upgrades of the core program and neatly separated data backup.

2. **Run Crucible on a dedicated machine, accessing its data on the local file system.**

   This is the best environment for swift Crucible performance. Avoid running Crucible in a virtual environment.

3. **Do not give Crucible projects the same key as your JIRA projects.**

---

Created by Atlassian in 2012. Licensed under a Creative Commons Attribution 2.5 Australia License.
When naming projects, take care to ensure that the key you assign to them is not the same as any of your JIRA projects. The reason for this is, if one of your Crucible projects has the same key as one of your projects in JIRA, then all links with that key will lead back to Crucible, rather than leading to JIRA, removing the ability to navigate between the two applications.

To avoid this, name your Crucible project keys differently. For example, you could place the following text at the beginning of each project key: \texttt{CR–} to distinguish it. So, for this case, if you have an existing JIRA key of \texttt{RHUBAB}, you would create a Crucible key called \texttt{CR-RHUBARB} so that they do not conflict.

4. \textbf{Do not use the built-in HSQLDB database for production use.}

The Crucible built-in database, running HSQLDB is somewhat susceptible to data loss during system crashes. We recommend that you do not use HSQLDB for production systems. External databases are generally more resistant to data loss during a system crash and are more suited for production use.

To see a list of external databases that Crucible supports, see the Supported Platforms page. For information on how to set up an external database, see the Crucible Database documentation.

\section*{Managing plugins}

A \textit{plugin} is an add-on to the core Crucible code, used to extend the Crucible functionality. Some plugins are shipped with Crucible, others are available for you to install yourself.

You can develop additional plugins (see the Crucible Developer docs) or download existing plugins from the Atlassian Plugin Exchange, and install them into your Crucible system.

\begin{itemize}
\item Managing your plugins
\item About the Universal Plugin Manager (UPM)
\item Notes
\end{itemize}

\section*{Managing your plugins}

- Viewing your Installed Plugins
- Installing a Plugin
- Configuring a Plugin
- Uninstalling a Plugin
- Disabling or Enabling a Plugin
- Checking Plugin Compatibility for Crucible Upgrades
- Upgrading your Existing Plugins
- Viewing the Plugin Audit Log

\section*{About the Universal Plugin Manager (UPM)}

The Universal Plugin Manager (UPM) provides a set of functions for managing your plugins. The UPM is itself a system plugin. It allows you to perform common tasks such as:

- Enabling and disabling plugins and their plugin modules.
- Installing new plugins.
- Configuring advanced plugin options.
- Finding out-of-date plugins and updating them.
- Checking the compatibility of your installed plugins against newer versions of the application.

Through the UPM you can interact with the Atlassian Plugin Exchange. You can use the UPM to browse available plugins for your application, and try or buy any of these plugins without ever leaving your application.

\textit{Screenshot: The Universal Plugin Manager in Crucible}
Notes

- **Plugin Safety.** Plugins are very powerful. They can change the behaviour of almost any part of the Crucible server. This makes it very important that you trust a plugin before you install it. Always be aware of where and who a plugin comes from.

- **Troubleshooting.** Having problems with the Universal Plugin Manager? Try the Universal Plugin Manager FAQ.

Viewing your Installed Plugins

Using the Universal Plugin Manager (UPM) you can see a list of plugins installed on your Crucible site. These include plugins that are bundled with Crucible as well as any plugins that you have installed. Both enabled and disabled plugins are displayed.

On this page:

- Viewing your Installed Plugins
- Viewing a Plugin's Details
- Notes

Viewing your Installed Plugins

To view your installed plugins:

1. Click the Manage Existing tab. You will see a list of the plugins installed in your application.
   - The plugins are grouped into 'User-installed Plugins' and 'System Plugins'.
   - You can filter your list by entering keywords in the 'Filter visible plugins' text box.
   - The list of 'System Plugins' will be hidden by default. Click the Show System Plugins link to see them.
   - Enabled plugins will have this icon: ★
   - Disabled plugins will have this icon: ✖
   - Click the name of a plugin to view the plugin's details.
   - Click Enable Safe Mode to run your application in safe mode. This mode disables all user installed plugins.

Viewing a Plugin's Details
You can view the details for a plugin by clicking the name of a plugin in the list of installed plugins. The summary contains a short description of the plugin as well as links to plugin operations and related information.

- **Plugin key** – A unique key that identifies each plugin in the system.
- **Developer** — The name of the plugin developer and a link to the developer's home page, if provided by the plugin developer.
- **Plugin version** – The version of the plugin currently installed.
- **Manage plugin modules** — Click this link to display the plugin modules below the plugin summary. A module is a component of the plugin. This link will only appear if the plugin has modules. If you want to enable or disable a plugin module, hover your mouse over the module and click the 'Enable'/Disable' button for that module.
- **Configure** – Click this link to display the configuration settings for the plugin. This link is disabled if the plugin is disabled. Please note that not all plugins have settings that can be configured through the Universal Plugin Manager.
- **Disable** – Click this button to disable the plugin in your application. This button will only appear if the plugin is enabled.
- **Enable** – Click this button to enable the plugin in your application. This button will only appear if the plugin is disabled.
- **Uninstall** – Click this button to uninstall the plugin from your application. This button will only appear for user installed plugins.

**Notes**

- **What is the difference between a 'system plugin' and a 'user installed plugin'?**
  - System plugins are shipped with the application. These plugins are essential for the functioning of the system. Although you can disable some of them, you should not do so unless instructed by an Atlassian support engineer. *Note: Not every system plugin can be disabled. You cannot uninstall any system plugins.*
  - User installed plugins are those which have been installed in the application after it was set up. You can install a plugin either by uploading a JAR file or via the Universal Plugin Manager. You can uninstall these plugins.

**Related Topics**

- Managing plugins
- Installing a Plugin

**Installing a Plugin**

This page describes how to install a plugin in Crucible. You can use plugins to customise and extend the functionality of your application.

A number of plugins are available from the Atlassian Plugin Exchange. You can also create your own as described in the Crucible developer's guide.

**On this page:**
- Adding a Plugin from the Atlassian Plugin Exchange
- Uploading your own plugin
- Notes

**Adding a Plugin from the Atlassian Plugin Exchange**

To access the Universal Plugin Manager in Crucible:

1. Log in as an administrator and go to the Crucible administration screen.
2. Click 'Plugins' in the left-hand menu.

To find a plugin in the Atlassian Plugin Exchange and add it to Crucible:
1. Click the 'Install' tab in the UPM. You will see a list of featured plugins.

2. Search for your plugin as follows:
   - Enter some keywords that describe the plugin in the 'Search the Plugin Exchange' search box and press 'Enter'.
   - Alternatively, browse to the desired plugin in the list. You can choose 'Featured', 'Popular', 'Supported' (by Atlassian) or 'All available' from the 'Plugins to show' dropdown to see a different list of plugins.

3. Click the 'Install' button for the desired plugin to add it to your application. A confirmation message and the plugin details will appear when the plugin is installed successfully.
   
   **Note:** You may need to restart your application for your change to take effect. The Universal Plugin Manager will inform you if this is the case.

   **Note:** Not all plugins can be automatically installed. Some required manual installation. These plugins will have a 'Download' button instead of an 'Install' button. In these cases, you should read and follow the plugin's installation instructions.

### Uploading your own plugin

#### To access the Universal Plugin Manager in Crucible:
1. Log in as an administrator and go to the Crucible administration screen.
2. Click 'Plugins' in the left-hand menu.

#### To upload your own plugin to Crucible:
1. Click the 'Install' tab in the UPM. You will see a list of featured plugins.
2. Click the 'Upload Plugin' link. The 'Upload Plugin' window will appear.
3. Specify the location of your plugin:
   - If the plugin you want to install is on your computer, use the 'Browse' dialogue to choose the plugin JAR file.
   - If you want to install a plugin from a remote location, enter the URL of the plugin JAR file in the 'From this URL' text box.
4. Click the 'Upload' button to upload and enable your plugin. A confirmation message will appear when the plugin is successfully installed.
   
   **Note:** You may need to restart your application for your change to take effect. The Universal Plugin Manager will inform you if this is the case.

**Screenshot: Uploading a new plugin**

![Upload Plugin](image)

**Notes**

- In Crucible, you can install and uninstall both version 1 and version 2 plugins using the Universal Plugin Manager. You will see an 'Install' or an 'Uninstall' button.
• Some entries that you find listed in the Universal Plugin Manager are not actually plugins. These entries will show a 'Download' button which allows you to download the application to your desktop and run it.

Related Topics
Managing plugins

Configuring a Plugin

A number of Crucible plugins offer advanced configuration options. If you have one of these plugins installed in your application, you can view and update these configuration options via the Universal Plugin Manager (UPM).

If you would like to disable or enable a plugin, please refer to Disabling or Enabling a Plugin.

To access the Universal Plugin Manager in Crucible:
  1. Log in as an administrator and go to the Crucible administration screen.
  2. Click 'Plugins' in the left-hand menu.

To configure a plugin in Crucible:
  1. Click the 'Manage Existing' tab.
  2. Locate the plugin that you want to configure and click its title. The plugin details section will expand.
  3. Click the 'Configure' link for that plugin.
     • The link will be disabled if the plugin is disabled.
     • If there is no 'Configure' link, then there are no advanced configuration options available for that plugin.
  4. The advanced configuration options for the plugin will appear. Update the configuration settings as desired and save your changes.

Note: The advanced configuration screens are provided by the plugin. If you encounter any problems after you click the 'Configure' link, the plugin is responsible for the issue, not the Universal Plugin Manager.

Related Topics
Managing plugins

Uninstalling a Plugin

If you wish to completely remove a plugin from Crucible, you can uninstall it via the Universal Plugin Manager (UPM). If you only want to temporarily remove it, you may wish to disable the plugin instead.

To access the Universal Plugin Manager in Crucible:
  1. Log in as an administrator and go to the Crucible administration screen.
  2. Click 'Plugins' in the left-hand menu.

To uninstall a plugin from Crucible:
  1. Click the 'Manage Existing' tab. You will see a list of the plugins installed in your application.
  2. Click the name of the plugin that you wish to uninstall. The plugin details will appear.
  3. Click the 'Uninstall' button. The information summary will display an 'Uninstalling' message and the plugin will be uninstalled from your application.

Related Topics
Managing plugins

Disabling or Enabling a Plugin

The Universal Plugin Manager (UPM) allows you to disable a plugin in Crucible without permanently removing it. You can also enable any plugins that have been previously disabled. If you want to add or remove a plugin from
your Crucible site, please refer to Installing a Plugin or Uninstalling a Plugin respectively.

You can also disable all user installed plugins in Crucible, by enabling safe mode. This may help you to diagnose a plugin-related problem more easily.

On this page:
- Disabling a Plugin
- Enabling a Plugin
- Disabling/Enabling all User Installed Plugins (Safe Mode)

Disabling a Plugin

To access the Universal Plugin Manager in Crucible:
1. Log in as an administrator and go to the Crucible administration screen.
2. Click 'Plugins' in the left-hand menu.

To disable a plugin in Crucible:
1. Click the 'Manage Existing' tab. You will see a list of the plugins installed in your application. Enabled plugins will have this icon: ⬤
2. Locate the plugin that you want to disable and click the title to expand the plugin details section.
3. Click the 'Disable' button.
4. Once a plugin has been disabled, you may need to restart your application for your change to take effect. If so, you will see a message for the plugin, 'Disabled, requires restart'. Once the plugin is fully disabled, you will see an 'Enable' link for the plugin.

Enabling a Plugin

To access the Universal Plugin Manager in Crucible:
1. Log in as an administrator and go to the Crucible administration screen.
2. Click 'Plugins' in the left-hand menu.

To enable a plugin in Crucible:
1. Click the 'Manage Existing' tab. You will see a list of the plugins installed in your application. Disabled plugins will have this icon: ⬤
2. Locate the plugin that you want to enable and click the title to expand the plugin details section.
3. Click the 'Enable' button.
4. Once a plugin has been enabled, you may need to restart your application for your change to take effect. If so, you will see a message for the plugin, 'Enabled, requires restart'. Once the plugin is fully disabled, you will see a 'Disable' link for the plugin.

Disabling/Enabling all User Installed Plugins (Safe Mode)

Running your application in safe mode disables all user installed plugins at once. By 'user installed plugins', we mean plugins that were not shipped with Crucible but were installed later via the UPM.

All plugins that were disabled when you entered safe mode will be re-enabled when you exit safe mode.

To access the Universal Plugin Manager in Crucible:
1. Log in as an administrator and go to the Crucible administration screen.
2. Click 'Plugins' in the left-hand menu.

To enable safe mode in Crucible:
1. Click the ‘Manage Existing’ tab. You will see a list of the plugins installed in your application.
2. Click the ‘Enable Safe Mode’ button.
3. Click the ‘Continue’ button in the confirmation window. All user installed plugins will be disabled and your application will now be running in ‘Safe Mode’.
4. You can now make changes to your installed plugins as desired. For example, you may want to enable/disable specific plugins or plugin modules.
5. Exit safe mode by clicking one of the links in the Safe Mode banner:
   - Click ‘Exit Safe Mode and restore the previous configuration’ to restore your plugin configuration to its state before you entered Safe Mode.
   - Click ‘Exit Safe Mode and keep the current configuration’ to keep all changes made to your plugin configuration during Safe Mode.

Related Topics

Managing plugins

Checking Plugin Compatibility for Crucible Upgrades

You can use the plugin manager’s ‘Upgrade Check’ to verify that your plugins will still work after a Crucible upgrade.

For example, if you are thinking of upgrading from Crucible 2.5 to Crucible 2.6, the upgrade check can tell you the following:

- Installed plugins that are compatible with Crucible 2.5 and Crucible 2.6.
- Installed plugins that are not compatible with Crucible 2.6 but will be compatible with Crucible 2.6 if you upgrade them.
- Installed plugins that are not compatible with Crucible 2.6, even if you upgrade them to their latest version.

To access the Universal Plugin Manager in Crucible:

1. Log in as an administrator and go to the Crucible administration screen.
2. Click ‘Plugins’ in the left-hand menu.

To check the compatibility of your plugins against different Crucible versions:

1. Click the ‘Upgrade Check’ tab.
2. In the ‘Check compatibility for’ dropdown menu, select the version of your application to check the plugins against.
3. Click the ‘Check’ button.
4. The page display any of your installed plugins that are not compatible with the selected application version. The compatibility checker will also check the compatibility of the latest available version of each plugin (if not already upgraded). You can click on the name of any of the plugins to view more information about the plugin.

The plugins are grouped into sections under the following headings:

- **Incompatible** – The installed versions of these plugins are not compatible with the selected application version. There are currently no plugin upgrades available that are compatible with the selected application version.
- **Compatible, if upgraded** – The installed versions of these plugins are not compatible with the selected application version. However, the plugins will be compatible if you upgrade them. There are buttons allowing you to upgrade these plugins.
- **Compatible if both <the application> and the plugin are upgraded** – The installed versions of these plugins are not compatible with the selected application version. There is a plugin compatible with the newer application version, but it is not compatible with the application version you are currently running. You must upgrade the application and then upgrade the plugin. There are buttons allowing you to disable these plugins before proceeding with the upgrade.
- **Compatible** – The currently installed versions of these plugins are compatible with the selected application version.
application version.

- **Unknown** – These plugins may or may not be compatible with the selected application version. If a plugin is not registered with the Atlassian Plugin Exchange, the Universal Plugin Manager cannot check its compatibility with different application versions.

Related Topics

Managing plugins

**Upgrading your Existing Plugins**

Plugins are often developed separately from Crucible. You may wish to upgrade your plugins to more recent versions to allow them to work with your Crucible version or simply to take advantage of new features in a plugin version. The Universal Plugin Manager (UPM) provides you with a list of plugins that have available upgrades and allows you to upgrade each plugin individually or in bulk.

On this page:

- Upgrading a Plugin
- Upgrading All your Plugins
- Notes

**Upgrading a Plugin**

**To access the Universal Plugin Manager in Crucible:**

1. Log in as an administrator and go to the Crucible administration screen.
2. Click 'Plugins' in the left-hand menu.

**To upgrade a plugin in Crucible:**

1. Click the 'Upgrade' tab. The plugin upgrades page will appear.
   - If there is a later version of a plugin that you have already installed, this page will show the latest compatible version of the plugin.
   - You can click the plugin name to expand the row and see more information about the plugin.
   - You can filter your list by entering keywords in the 'Filter plugins' text box.
2. Click the 'Upgrade Now' button next to the relevant plugin to update it to the plugin version shown.

**Upgrading All your Plugins**

**To access the Universal Plugin Manager in Crucible:**

1. Log in as an administrator and go to the Crucible administration screen.
2. Click 'Plugins' in the left-hand menu.

**To upgrade all available plugins in Crucible:**

1. Click the 'Upgrade' tab. The plugin upgrades page will appear.
   - If there is a later version of a plugin that you have already installed, this page will show the latest compatible version of the plugin.
   - You can click the plugin name to expand the row and see more information about the plugin.
   - You can filter your list by entering keywords in the 'Filter plugins' text box.
2. Click the 'Upgrade All' button to update every plugin to the plugin versions shown.

*Note:* Some plugins cannot be installed via the Universal Plugin Manager. You must install these plugins manually. These plugins will not be upgraded automatically.

**Notes**

- If you are considering upgrading Crucible, you can use the Universal Plugin Manager to check the compatibility of your plugins with your desired Crucible version. See Checking Plugin Compatibility for Crucible Upgrades.

Created by Atlassian in 2012. Licensed under a Creative Commons Attribution 2.5 Australia License.
Related Topics

Managing plugins

Viewing the Plugin Audit Log

The Universal Plugin Manager (UPM) keeps a log of all plugin activity for your Crucible site. Such activities may be adding plugins, enabling plugins, and so on. You can adjust the period of time for which log entries are kept.

On this page:
- Viewing the Plugin Audit Log
- Configuring the Plugin Audit Log

Viewing the Plugin Audit Log

To access the Universal Plugin Manager in Crucible:
1. Log in as an administrator and go to the Crucible administration screen.
2. Click ‘Plugins’ in the left-hand menu.

To view the plugin audit log:
1. Click the ‘Audit Log’ tab. The plugin audit log will appear, showing the 25 most recent entries.
2. Use the arrows if you want to view older entries.
3. Click the orange RSS icon if you want to receive the audit log activity in an RSS feed.

Configuring the Plugin Audit Log

To access the Universal Plugin Manager in Crucible:
1. Log in as an administrator and go to the Crucible administration screen.
2. Click ‘Plugins’ in the left-hand menu.

To configure the length of time for which log entries are kept:
1. Click the ‘Audit Log’ tab. The plugin audit log will appear.
2. Click the ‘Configure purge policy’ link.
3. In the ‘Purge audit log after’ field, specify the number of days for which you wish to keep the logs.
4. Click the ‘Confirm’ button.

Related Topics

Managing plugins

Migrating to an External Database

This page contains information about migrating Crucible from its default embedded HSQL database to an external database. Advantages of using a database other than the embedded HSQL database include:

- **Improved Protection Against Data Loss:** The Crucible built-in database, running HSQLDB is somewhat susceptible to data loss during system crashes. External databases are generally more resistant to data loss during a system crash. HSQLDB is not supported in production environments and should only be used for evaluation purposes.

- **Performance & Scalability:** if you have many users on your Crucible instance, running the database on the same server as FishEye may slow it down. When using the embedded database, the database will always be hosted and run on the same server as Crucible.

- **Data Stored in the Crucible Database:** The Crucible database stores all information besides the cache for repository scans. This means all reviews, comments, review states, user data and user preferences information.

On this page:
Supported Databases

You can use a number of alternatives to the built-in HSQLDB database for storing FishEye and Crucible's relational data. The supported alternative databases are listed on the Supported Platforms page. Please note, that only the database versions listed on that page are supported.

The pages linked below outline the steps required to switch to an external database:

- Migrating to MySQL Enterprise Server
- Migrating to Oracle
- Migrating to PostgreSQL
- Migrating to SQL Server

Support for Other Databases

If you are using another database product that you would like to see supported, please create a JIRA issue for it under the Crucible project.

Notes

Crucible uses Read Committed transaction isolation. There is no requirement to configure this explicitly when setting up an external database - Crucible will configure the transaction isolation when connecting to the database.

Migrating to MySQL Enterprise Server

To switch to a MySQL Enterprise Server, install MySQL Enterprise Server and then follow the steps below. Please note that during the migration of database servers, the FishEye/Crucible instance will not be available to users or to external API clients.

On this page:

- Step 1. Install and Create a MySQL Database
- Step 2. Configure FishEye/Crucible to use MySQL, and Migrate Data

Related pages:

- Migrating to PostgreSQL
- Migrating to Oracle
- Migrating to SQL Server
- Migrating to an External Database
- Troubleshooting Databases

Step 1. Install and Create a MySQL Database

Before you begin:

1. Ensure you install a version of MySQL that is supported. Check Supported platforms for exact versions.
2. The JDBC drivers for MySQL Enterprise Server are bundled with FishEye/Crucible. Skip to step 2 if this meets your needs. If you want to install a specific, different version of the bundled JDBC driver, download the MySQL Enterprise Server JDBC driver (a .jar file) from the download website and copy the .jar file
3. Create a UTF-8 Database:

```
CREATE DATABASE fisheye CHARACTER SET utf8 COLLATE utf8_bin;
```

4. You will also need to set the Server Character set to utf8. This can be done by adding the following in `my.ini` for Windows or `my.cnf` for other operating systems (create the file at `/etc/my.cnf` if it doesn’t already exist). It has to be declared in the Server section, which is the section after `[mysqld]`:

```
[mysqld]
character-set-server=utf8
```

5. Use the `status` command to verify database character encoding information.
First select the `fisheye` database:
```
mysql> use fisheye;
```
Then issue the `mysql> status;` command:

```
mysql> status
---------------------
/Applications/xampp/xamppfiles/bin/mysql Ver 14.12 Distrib 5.0.67, for apple-darwin0.11.1 (i686) using EditLine wrapper

Connection id: 5754
Current database: fisheye
Connection user: root@localhost
SSL: Not in use
Current pager: stdout
Using outfile: ...
Using delimiter: ;
Server version: 5.0.67 Source distribution
Protocol version: 10
Connection: Connect via UNIX socket
Server character set: utf8
Db character set: utf8
Collation: latin1
UNIX socket: /Applications/xampp/xamppfiles/var/mysql/mysql.sock
Uptime: 29 days 1 hour 24 min 56 sec

Threads: 6 Questions: 49238134 Slow queries: 0 Opens: 460445 Flush tables: 1
Open tables: 4 Queries per second avg: 19.607
```

6. Create a user that can log in from the host that FishEye/Crucible is running on and make sure that the user has full access to the newly created database. In particular, the user should be allowed to create and drop tables, indexes and other constraints.

For instance, when FishEye/Crucible and MySQL Enterprise Server run on the same machine (accessible through `localhost`), issue the following commands (replacing username and password with the appropriate values):
mysql> grant all on fisheye.* to 'username'@'localhost' identified by 'password';
Query OK, 0 rows affected (0.00 sec)

mysql> flush privileges;
Query OK, 0 rows affected (0.01 sec)

**Step 2. Configure FishEye/Crucible to use MySQL, and Migrate Data**

In order to migrate to a different database backend, you must create a backup of sql data, configure the database and finally import the data via a backup restoration process. This can be done from either the FishEye/Crucible administration console, which streamlines the process, or via the command line tool which FishEye/Crucible provides.

**From FishEye/Crucible’s Administration**

1. Navigate to the Database page in FishEye/Crucible’s Administration console
2. Then click ‘Test Connection’ to verify that FishEye/Crucible can log in to the database:
3. Select MySQL from the database type
4. Fill in the appropriate fields, replacing the host, port, database name, username and password as required
5. Click on Test Connection to validate the values

**Screenshot: Testing the Connection**

If this fails, verify that you have the MySQL Enterprise Server JDBC driver `.jar` file in the classpath (by placing the `.jar` file in `FISHEYE_INST/lib`). Also, ensure that the database user can log in to the database from the machine that FishEye/Crucible is running on and that all the required privileges are present.

6. Click ‘Save & Migrate Data’ to start the migration process.

During the migration process (which will take several minutes, depending on the size of your database and network throughput), the product will be inaccessible to users and external API clients. Users will see a maintenance screen that informs them of the process. Should the migration fail for any reason, FishEye/Crucible will not switch to the new database and report on the encountered problems. Because the destination database may now contain some, but not yet all data, drop all tables, indexes and
constraints before attempting a new migration.

**Screenshot: Migrating the Database**

Migrating Database

Migrating database...

From: jdbc:h2:file:/Users/ervijst/centena/demotion/var/data/crudb/crucible
To: jdbc:mysql://localhost/crucible

Started
Creating table definitions...
1 rows written, 0 tables completed.
14 rows written, 4 tables completed.
2328 rows written, 2 tables completed.
7327 rows written, 2 tables completed.
11143 rows written, 3 tables completed.
12676 rows written, 3 tables completed.
12740 rows written, 4 tables completed.
22739 rows written, 4 tables completed.
30458 rows written, 5 tables completed.
39493 rows written, 5 tables completed.
39497 rows written, 6 tables completed.

From the command line
1. Create a backup of the sql data from the FishEye/Crucible instance. Information on how to create a backup can be found at [Backing Up and Restoring FishEye Data](#) 
2. Run the following command from the bin directory in FISHEYE_INST

```
$ ./fisheyectl.sh restore --sql \
   --file /path/to/backup.zip \
   --dbtype mysql \
   --jdbcurl jdbc:mysql://hostname/dbname \
   --username crucible \
   --password password
```

3. When the import is complete, FishEye/Crucible can be started and will use MySQL.

**Migrating to Oracle**

To switch to an Oracle database, install Oracle and follow the steps below. Please note that during the migration of database servers, the FishEye/Crucible instance will not be available to users or to external API clients.

Oracle support for FishEye/Crucible and Crucible was introduced in version 2.5.0. In order to migrate to Oracle, your instance must be currently running at least version 2.5. If you are running an older version, then you will be required to first upgrade FishEye/Crucible and then migrate.
On this page:

- Step 1. Install and Create a Oracle Database
- Step 2. Configure FishEye/Crucible to use Oracle, and Migrate Data

Related pages:

- Migrating to MySQL Enterprise Server
- Migrating to PostgreSQL
- Migrating to SQL Server
- Migrating to an External Database
- Troubleshooting Databases

Step 1. Install and Create a Oracle Database

1. The JDBC drivers for Oracle are bundled with FishEye/Crucible. Skip to step 2 if this meets your needs. If you want to install a specific, different version of the bundled JDBC driver, download the Oracle JDBC driver [file](http://www.oracle.com/technetwork/database/features/jdbc/index-091264.html) and copy the .jar file to your FISHEYE_INST/lib directory (create the lib/ directory if it doesn't already exist). Move the existing JDBC .jar file to another location (and back it up). Restart FishEye/Crucible to have it pick up the new driver.

2. Because creating a database with Oracle is a complex process, we recommend speaking to your resident DBA for creation of a new database for usage with Crucible. We highly recommend installing Oracle with the AL32UTF8 encoding otherwise you may see encoding issues in the product.

Step 2. Configure FishEye/Crucible to use Oracle, and Migrate Data

In order to migrate to a different database backend, you must create a backup of sql data, configure the database and finally import the data via a backup restoration process. This can be done from either the FishEye/Crucible administration console, which streamlines the process, or via the command line tool which FishEye/Crucible provides.

From FishEye/Crucible’s Administration

1. Navigate to the Database page in FishEye/Crucible’s Administration console
2. Then click ‘Test Connection’ to verify that FishEye/Crucible can log in to the database:
3. Select Oracle from the database type
4. Fill in the appropriate fields, replacing the host, port, database name, username and password as required
5. Click on Test Connection to validate the values

```
Screenshot: Testing the Connection
```
If this fails, verify that you have the Oracle JDBC driver `.jar` file in the classpath (by placing the `.jar` file in `FISHEYE_INST/lib`). Also, ensure that the database user can log in to the database from the machine that FishEye/Crucible is running on and that all the required privileges are present.

6. Click ‘Save & Migrate Data’ to start the migration process.

During the migration process (which will take several minutes, depending on the size of your database and network throughput), the product will be inaccessible to users and external API clients. Users will see a maintenance screen that informs them of the process. Should the migration fail for any reason, FishEye/Crucible will not switch to the new database and report on the encountered problems. Because the destination database may now contain some, but not yet all data, drop all tables, indexes and constraints before attempting a new migration.

**Screenshot: Migrating the Database**

1. Create a backup of the `sql` data from the FishEye/Crucible instance. Information on how to create a
backup can be found at Backing Up and Restoring FishEye Data \ Backing Up and Restoring Crucible Data

2. Run the following command from the bin directory in FISHEYE_INST

```bash
$ ./fisheyectl.sh restore --sql
--file /path/to/backup.zip
--dbtype oracle
--jdbcurl jdbc:oracle:thin:@hostname:port:dbname
--username crucible
--password password
```

3. When the import is complete, FishEye/Crucible can be started and will use Oracle.

**Migrating to PostgreSQL**

To switch to a PostgreSQL database, install PostgreSQL and follow the steps below. Please note that during the migration of database servers, the FishEye/Crucible instance will not be available to users or to external API clients.

**On this page:**
- Step 1. Install and Create a PostgreSQL Database
- Step 2. Configure FishEye/Crucible to use PostgreSQL, and Migrate Data

**Related pages:**
- Migrating to MySQL Enterprise Server
- Migrating to Oracle
- Migrating to SQL Server
- Migrating to an External Database
- Troubleshooting Databases

**Step 1. Install and Create a PostgreSQL Database**

1. The JDBC drivers for PostgreSQL are bundled with FishEye/Crucible. Skip to Step 2 if this meets your needs. If you want to install a specific, different version of the bundled JDBC driver, download the PostgreSQL JDBC driver .jar file from the PostgreSQL website and copy the .jar file to your FISHEYE_INST/lib directory (create the lib/ directory if it doesn't already exist). Move the existing JDBC .jar file to another location (and back it up). Restart FishEye/Crucible to have it pick up the new driver.

2. Create a new database user (replacing 'username' and 'password' with the appropriate values):

   ```bash
   $ psql
   > create user username password 'password';
   ```

3. Create a UTF-8 database and make the newly created user the owner:

   ```bash
   > create database crucible ENCODING 'UTF-8' OWNER username;
   ```

4. Make sure the user has full access to the database:
Step 2. Configure FishEye/Crucible to use PostgreSQL, and Migrate Data

In order to migrate to a different database backend, you must create a backup of sql data, configure the database and finally import the data via a backup restoration process. This can be done from either the FishEye/Crucible administration console, which streamlines the process, or via the command line tool which FishEye/Crucible provides.

From FishEye/Crucible’s Administration

1. Navigate to the Database page in FishEye/Crucible's Administration console
2. First click 'Edit' and then click 'Test Connection' to verify that FishEye/Crucible can log in to the existing database
3. Select PostgreSQL from the database Type
4. Fill in the appropriate fields:
   a. Driver Location: either your own PostgreSQL JDBC or the Bundled one that came with FishEye
   b. URL: create this field by replacing the host, port, and database name with your own (i.e. jdbc:postgresql://localhost:5432/<dbname> e.g. jdbc:postgresql://localhost:5432/crucible)
   c. Username: your DB username
   d. Password: your DB password
   e. Minimum Pool Connections: 5 is the default
   f. Maximum Pool Connections: 20 is the default
   g. Parameters: (one per line)
      i. useUnicode=true
      ii. characterEncoding=UTF8

5. Click on Test Connection to validate the values

Screenshot: Testing the Connection

To switch to a different database, specify the database's configuration settings in the form below and use the Test Connection button to verify that the database can be used. We are currently connected to the PostgreSQL database at jdbc:postgresql://localhost:5432/crucible.

If this fails, verify that you have the PostgreSQL JDBC driver .jar file in the classpath (by placing the .jar file in FISHEYE_INST/lib). Also, ensure that the database user can log in to the database from the machine that FishEye/Crucible is running on and that all the required privileges are present.

6. Click 'Save & Migrate Data' to start the migration process.
During the migration process (which will take several minutes, depending on the size of your database and network throughput), the product will be inaccessible to users and external API clients. Users will see a maintenance screen that informs them of the process. Should the migration fail for any reason, FishEye/Crucible will not switch to the new database and report on the encountered problems. Because the destination database may now contain some, but not yet all data, drop all tables, indexes and constraints before attempting a new migration.

**Screenshot: Migrating the Database**

From the command line

1. Create a backup of the sql data from the FishEye/Crucible instance. Information on how to create a backup can be found at [Backing Up and Restoring FishEye Data](#) / [Backing Up and Restoring Crucible Data](#)
2. Run the following command from the bin directory in FISHEYE_INST

   ```bash
   $ ./fisheyectl.sh restore --sql \
   --file /path/to/backup.zip \
   --dbtype postgresql \
   --jdbcurl jdbc:postgresql://hostname/dbname \
   --username crucible \
   --password password
   ```
3. When the import is complete, FishEye/Crucible can be started and will use PostgreSQL

**Migrating to SQL Server**

To migrate FishEye/Crucible to an SQL Server database, install SQL Server and follow the steps below.
Before You Begin

- Check that you are using version of SQL Server that is supported for use with FishEye. See Supported platforms.

An existing Java bug prevents connection with Java 1.6.0_29 and above (including Java 1.7.0). Read more about the issue and possible workarounds here.

Step 1. Install and Create an SQL Server Database

See the SQL Server Online resources (MSDN) for instructions on how to install and create an SQL Server database.

Please note the following FishEye/Crucible-specific information when installing and creating an SQL Server database:

- The JDBC jtds drivers for SQLServer are bundled with Crucible. We do not support using the Microsoft distributed jdbc driver.
- Your database server must be configured to use the Latin1_General_CS_AS collation set.

Step 2. Configure FishEye/Crucible to Use SQL Server and Migrate Data

In order to migrate to a different database backend, you must create a backup of SQL data, configure the database and finally import the data via a backup restoration process. This can be done from either the Crucible administration console, which streamlines the process, or via the command line tool which Crucible provides. These two methods are described below. The following resources may be of interest:

- Backing Up and Restoring FishEye Data
- Backing Up and Restoring Crucible Data
- SQL Server Online resources (MSDN)

Configuring and Migrating via FishEye/Crucible's Administration Console

Before you begin:

- Note, during the migration process (which will take several minutes, depending on the size of your database and network throughput), the FishEye/Crucible instance will be inaccessible to users and
external API clients. Users will see a maintenance screen that informs them of the process.

- If you are attempting a migration after a previous migration has failed, you must drop all tables, indexes and constraints before attempting a new migration. This is because the destination database may contain data from the previous migration attempt.
- Verify that you have the jtds JDBC driver .jar file in the classpath (by placing the .jar file in FISHEYE_INST/lib).
- Ensure that the database user can log in to the database from the machine that FishEye/Crucible is running on and that all the required privileges are present.

To configure FishEye/Crucible to use SQL Server and migrate data using the Administration Console:

1. Navigate to the ‘Database’ page in FishEye/Crucible’s Administration console.
2. Configure FishEye/Crucible to use SQL Server, as follows:
   - Select ‘SQLServer’ from the ‘Type’ dropdown.
   - Complete the appropriate fields, replacing the ‘URL’ (host, port and database name), ‘User Name’ and ‘Password’ as required.
     NOTE: The default SQL server instance listens on port 1433. If your instance is not the default, use the port number that is associated with your particular instance.
     e.g. URL: jdbc:jtds:sqlserver://localhost:1433;databaseName=your database name here;
3. Click ‘Test Connection’ to verify that FishEye/Crucible can log in to the database (see ‘Testing the Connection’ screenshot below).
4. Click ‘Save & Migrate Data’ to start the migration process (see ‘Migrating the Database’ screenshot below). If the migration fails, FishEye/Crucible will not switch to the new database and will report the problems encountered.

Configuring and Migrating via the Command Line

To configure FishEye/Crucible to use SQL Server and migrate data using the Command Line:

1. Create a backup of the sql data from the FishEye/Crucible instance. Information on how to create a backup can be found at Backing Up and Restoring FishEye Data / Backing Up and Restoring Crucible Data
2. Run the following command from the bin directory in FISHEYE_INST:
3. When the import is complete, FishEye/Crucible can be started and will use SQLServer.

[ Configuring and Migrating via FishEye/Crucible's Administration Console ]  [ Configuring and Migrating via the Command Line ]

Integrating Crucible with Other Applications

- Configuring Application Links
- JIRA Integration in Crucible
- Trusted Applications

Configuring Application Links

An application link is a trust relationship between two applications. Linking two applications allows you to share information and to access one application's functions from within the other.

For example, you may want to set up a trust relationship between Crucible and two JIRA servers.

![Application links to two JIRA servers](image)

Screenshot above: Application links to two JIRA servers

Related Topics

- Adding an Application Link
- Configuring Authentication for an Application Link
- Configuring Project Links across Applications
- Deleting an Application Link
- Editing an Application Link
- Making an Application Link the Primary Link
- Relocating an Application Link
- Upgrading an Application Link

Adding an Application Link

This page describes how to add a new application link in Crucible. The process for adding an application link is different depending on whether or not the application you are linking Crucible to supports Atlassian's Application Links.

If you are linking Crucible to an application that does not have Application Links, you will need to do additional configuration in that application. This is because Application Links in Crucible will not be able to automatically configure authentication in your remote application.
On this page:

- **Before You Begin**
- **Adding an Application Link to an Application that Supports Application Links**
- **Adding an Application Link to an Application that Does Not Support Application Links**

### Before You Begin

Check the following settings. This is required for synchronisation to work correctly:

- Make sure that the base URL is set correctly in Crucible. See the [FishEye documentation on configuring the web server](https://confluence.atlassian.com/display/FISH/Eye+doc+on+configuring+the+web+server).
- Make sure that the base URL is set correctly in the application which you intend to link to. See the appropriate instructions for **JIRA**, **FishEye**, **Bamboo**, **Confluence**.

### Adding an Application Link to an Application that Supports Application Links

**To link to an application that supports Application Links:**

1. Log in as a system administrator and go to the administration page. Click ‘Application Links’ in the administration menu. The ‘Configure Application Links’ page will appear, showing the application links that have been set up.
2. Click ‘Add Application Link’. Step 1 of the link wizard will appear.
3. Enter the **server URL** of the application that you want to link to (the ‘remote application’).
4. Click the ‘Next’ button. Step 2 of the link wizard will appear.
5. Enter the following information:
   - ‘Create a link back to this server’ – Tick this check box if you want to create a two-way link between the remote application and your application. If you want to do this, you will need to enter the username and password of an administrator for the remote application. **Note:** These credentials are only used to authenticate you to the remote application, so that Application Links can make the changes required for the new link. The credentials are not saved.
   - ‘Reciprocal Link URL’ – The URL you give here will override the base URL specified in your remote application's administration console, for the purposes of the application links connection. Application Links will use this URL to access the remote application.
6. Click the ‘Next’ button. Step 3 of the link wizard will appear.
7. Enter the information required to configure authentication for your application link:
   - ‘The servers have the same set of users’ or ‘The servers have different sets of users’ – Select one of these options depending on how you manage users between the two applications.
   - ‘These servers fully trust each other’ – Tick this check box if you know that the code in both applications will behave itself at all times and are sure each application will maintain the security of its private key. **For more information about configuring authentication, see Configuring Authentication for an Application Link.**
8. Click the ‘Create’ button to create the application link.
Adding an Application Link to an Application that Does Not Support Application Links

To link to an application that does not support Application Links:

1. Log in as a system administrator and go to the administration page. Click 'Application Links' in the administration menu. The 'Configure Application Links' page will appear, showing the application links that have been set up.
2. Click 'Add Application Link'. Step 1 of the 'Link to another server' dialogue will be displayed. 
3. Enter the server URL of the application that you want to link to, in the 'Server URL' field. Click the 'Next' button. Step 2 of the 'Link to another server' dialogue will be displayed.
4. Fill out the fields, as follows:
   - 'Application Name' — Enter the name by which this remote application will be referred to, in your application.
   - 'Application Type' — Select the type of application that you are linking to: Bamboo, FishEye/Crucible, JIRA, Confluence, Subversion.
   - 'Application URL' — This will be set to the server URL you entered in the previous step and will not be editable.
5. Click the 'Create' button to create the application link. The 'Configure Application Links' page will be displayed, listing all of the application links that have currently been set up for your application including the one you just added.
6. Configure the desired authentication type (Trusted Applications, OAuth, basic HTTP, none) for your new application link. See Configuring Authentication for an Application Link.
7. In your application that does not support Application Links, configure the same type of authentication that you configured for your application link's outgoing authentication (in the previous step). For example, if you configured outgoing Trusted Applications authentication in your Application-Links-enabled application, you also need log into your non-Application-Links application and manually configure Trusted Applications (see the relevant administrator's documentation for the application).
Choosing Authentication for an Application Link

The level of authentication that you should configure for your application link depends on a number of factors.

- Do the two applications trust each other? In other words, are you sure that the code in the application will behave itself at all times and that the application will maintain the security of its private key?
- Do the two applications share the same user base?
- Do you have administrative access to the application you are linking to?

Common scenarios include:

- If the two applications you are linking **trust each other and share the same user base**, configure **two-way authentication using Trusted Applications** for both incoming and outgoing authentication. For example, you may link your internal Crucible server to an internal JIRA server.
- If the two applications you are linking **trust each other but do not share the same user base**, configure **two-way authentication using OAuth** for both incoming and outgoing authentication. For example, you may link your internal Crucible server to an external (customer-facing) JIRA server.
- If you **do not have administrative rights to the application that you are linking to** (for example, linking to a public Crucible server), configure a **one-way outgoing link** authenticated using **basic HTTP**.
authentication or do not configure any authentication for the link. For example, you may link your external Crucible server to a partner organisation’s Crucible server. An unauthenticated link will still allow the local application to render hyperlinks to the remote application or query anonymously-accessible APIs.

The flowchart below provides a guide to what authentication you should configure for your application link.

Read the following topics for information on how to configure authentication for an application link:

- Configuring Basic HTTP Authentication for an Application Link
- Configuring OAuth Authentication for an Application Link
- Configuring Trusted Applications Authentication for an Application Link
- Incoming and Outgoing Authentication

Flowchart above: Determining what authentication to configure for an Application Link

Security Implications for each Authentication Type

If you configure Trusted Applications authentication for your application (meaning that your servers have the same set of users and they fully trust each other) please be aware of the following security implications:

- Trusted applications are a potential security risk. When you configure Trusted Applications authentication, you are allowing one application to access another as any user. This allows all of the built-in security measures to be bypassed. Do not configure a trusted application unless you know that all code in the application you are trusting will behave itself at all times, and you are sure that the application will maintain the security of its private key.

If you configure OAuth authentication for your application (meaning that your servers have different sets of users and they fully trust each other) please be aware of the following security implications:

- Adding an OAuth consumer requires the transmission of sensitive data. To prevent ‘man-in-the-middle’ attacks, it is recommended that you use SSL for your applications while configuring OAuth authentication.
• Do not link to an application using OAuth authentication, unless you trust all code in the application to behave itself at all times. OAuth consumers are a potential security risk to the applications that they are linked to.

About Primary Authentication Types

You can configure multiple authentication types for each application link. When a feature makes a request using an Application Link, it will use one of the configured authentication types. If more than one authentication type is configured, it will by default use the authentication type that is marked as the primary authentication type. The default authentication type is indicated by the green tick next to the authentication type on the list application link screen.
You cannot configure which authentication type is the primary authentication type. The primary authentication type is determined automatically by Application Links and depends on a weight defined by each authentication type method. However, every feature that uses Application Links can also choose to use a specific authentication type and might not use the default primary authentication type.

About Impersonating and Non-Impersonating Authentication Types

Applications Links allows you to configure 'impersonating' and 'non-impersonating' authentication types:

- **Impersonating authentication types** make requests on behalf of the user who is currently logged in. People will see only the information that they have permission to see. This includes OAuth and Trusted Applications authentication.

- **Non-impersonating authentication types** always use a pre-configured user when making a request. Everyone logged into the system will see the same information. This includes basic HTTP authentication.

Configuring Basic HTTP Authentication for an Application Link

This page describes how to configure Basic HTTP authentication for incoming and/or outgoing authentication for an application link.

Basic HTTP authentication allows Crucible to provide a specific set of user credentials to a remote application and vice versa. Once authenticated, one application can access specified functions on the other application on behalf of that user. For example, if you supply the credentials of a Crucible administrator on your Crucible server to a remote application, the remote application will be able to access all functions on your Crucible server that the Crucible administrator can access.

This method of authentication relies on the connection between Crucible and the remote application being secure. We recommend that you use Trusted Applications authentication or OAuth authentication for your application link instead, if possible.

**On this page:**
- Before You Begin
- Configuring Basic HTTP Authentication for Outgoing Authentication
- Configuring Basic HTTP Authentication for Incoming Authentication

**Before You Begin**

- The instructions assume that both of the applications that you are linking have the Application Links plugin installed. If the remote application that you are linking to supports Basic HTTP authentication, but does not have the Application Links plugin installed, you will need to configure Basic HTTP authentication from within the remote application. See the relevant administrator's documentation for the application. This is in addition to configuring the outgoing/incoming authentication for the application link described below.
- You must be a Crucible administrator to configure Basic HTTP authentication for an application link.

**Configuring Basic HTTP Authentication for Outgoing Authentication**

Configuring outgoing basic http authentication will allow Crucible to trust a remote application. This allows the remote application to access specified functions in Crucible.

**To configure basic http authentication for an outgoing application link:**

1. Log in as a system administrator and go to the administration page. Click 'Application Links' in the administration menu. The 'Configure Application Links' page will appear, showing the application links that have been set up.
2. Click the 'Configure' link next to the application link that you want to configure authentication for.
3. Click the 'Outgoing Authentication' tab. The outgoing authentication page will be displayed.
4. Click the 'Basic Access' tab.
5. Click the 'Configure' button and enter the credentials (username and password) that the remote application will use to log into your application.
6. Click the 'Apply' button to save your changes.

Configuring Basic HTTP Authentication for Incoming Authentication

Configuring incoming basic http authentication will allow the remote application that you are linking to, to trust Crucible. Crucible will be able to access specified functions on the remote application.

To configure basic http authentication for an incoming application link:

1. Log in as a system administrator and go to the administration page. Click 'Application Links' in the administration menu. The 'Configure Application Links' page will appear, showing the application links that have been set up.
2. Click the 'Configure' link next to the application link that you want to configure authentication for.
3. Click the 'Incoming Authentication' tab. The incoming authentication page will be displayed.
4. Click the 'Basic Access' tab.
5. Click the 'Configure' button and enter the credentials (username and password) that the your application will use to log in to the remote application.
6. Click the 'Apply' button to save your changes.

Related Topics

Configuring OAuth Authentication for an Application Link
Configuring Trusted Applications Authentication for an Application Link

Configuring OAuth Authentication for an Application Link

This page describes how to configure OAuth for outgoing authentication and/or incoming authentication for an application link.

OAuth is a protocol that allows a web application to share data/resources with any other OAuth-compliant application. These applications could be another web application (such as a JIRA site or an iGoogle home page), a desktop application or a mobile device application, provided that they are accessible from within your network or available on the Internet.

A typical scenario is to set up an application link between two applications which trust each other, have the Application Links plugin installed, but do not share the same set of users. In this case, you would configure OAuth for both outgoing and incoming authentication.

Key OAuth Terminology:

- **Service provider** – An application that shares ('provides') its resources.
- **Consumer** – An application that accesses ('consumes') a service provider's resources.
- **User** – An individual who has an account with the service provider.

On this page:

- Before You Begin
- Configuring OAuth for Outgoing Authentication
- Configuring OAuth for Incoming Authentication

Related pages:

- Configuring Basic HTTP Authentication for an Application Link
- Configuring Trusted Applications Authentication for an Application Link

Created by Atlassian in 2012. Licensed under a Creative Commons Attribution 2.5 Australia License.
Before You Begin

- Adding an OAuth consumer requires the transmission of sensitive data. To prevent 'man-in-the-middle' attacks, it is recommended that you use SSL for your applications while configuring OAuth authentication.
- Do not link to an application using OAuth authentication, unless you trust all code in the application to behave itself at all times. OAuth consumers are a potential security risk to the applications that they are linked to.
- The instructions assume that both of the applications that you are linking have the Application Links plugin installed. If the remote application supports OAuth but does not have the Application Links plugin installed, you will need to configure OAuth from within the remote application (see the administrator's documentation for the application) in addition to configuring the outgoing/incoming authentication for the application link as described below.
- You must be a Crucible administrator to configure OAuth authentication for an application link.

Configuring OAuth for Outgoing Authentication

Configuring outgoing OAuth authentication will allow Crucible to access data in a remote application on behalf of a user. In other words, Crucible can access specified functions in the remote application.

To configure OAuth authentication for an outgoing application link:

1. Log in as a system administrator and go to the administration page. Click 'Application Links' in the administration menu. The 'Configure Application Links' page will appear, showing the application links that have been set up.
2. Click the 'Configure' link next to the application link that you want to configure OAuth for.
3. Click the 'Outgoing Authentication' tab. The outgoing authentication page will be displayed.
4. Click the 'OAuth' tab.
5. If you are not currently logged in to the remote application (or you logged in to the remote application under a variant of the application's hostname, such as the IP address), a login dialogue will display.
   - Enter the 'Username' and 'Password' for the remote server, not your local server, and click the 'Login' button. The remote server needs to learn the identity of your local server for the OAuth protocol to work and your admin credentials are used to store your local server's public key on the remote server. If you are already logged into your remote server, then the appropriate changes can be made without having to log in again.
6. Click the 'Enable' button to enable OAuth authentication for the outgoing link. Your application will be automatically set up to be the 'consumer' and the remote application as a 'service provider'.

Configuring OAuth for Incoming Authentication

Configuring incoming OAuth authentication will allow the remote application to access data in Crucible.

To configure OAuth authentication for an incoming application link:

1. Log in as a system administrator and go to the administration page. Click 'Application Links' in the administration menu. The 'Configure Application Links' page will appear, showing the application links that have been set up.
2. Click the 'Configure' link next to the application link that you want to configure OAuth for.
3. Click the 'Incoming Authentication' tab. The incoming authentication page will be displayed.
4. Click the 'OAuth' tab.
5. Click the 'Enable' button to enable OAuth authentication for the incoming link. The remote application will be automatically set up to be the 'consumer' and your local application as a 'service provider'.

Configuring Trusted Applications Authentication for an Application Link

This page describes how to configure Trusted Applications for outgoing and/or incoming authentication for an application link.
Trusted Applications authentication allows one application to gain access to specified functions within another application on behalf of any user, without the user having to log in to the second application.

A typical scenario is to set up an application link between two applications which trust each other, have the same set of users and have the application links plugin installed. In this case, you would configure Trusted Applications for both outgoing and incoming authentication.

On this page:
- Before You Begin
- Configuring Trusted Applications for Outgoing Authentication
- Configuring Trusted Applications for Incoming Authentication

Before You Begin
- Trusted applications are a potential security risk. When you configure Trusted Applications authentication, you are allowing one application to access another as any user. This allows all of the built-in security measures to be bypassed. Do not configure a trusted application unless you know that all code in the application you are trusting will behave itself at all times, and you are sure that the application will maintain the security of its private key.
- The instructions below assume that both of the applications that you are linking have the Application Links plugin installed. If the remote application supports Trusted Applications but does not have the Application Links plugin installed, you will need to configure Trusted Applications from within the remote application (see the relevant administrator's documentation for the application) in addition to configuring the outgoing/incoming authentication for the application link as described below.
- You must be a Crucible administrator to configure Trusted Applications authentication for an application link.

Configuring Trusted Applications for Outgoing Authentication

Configuring outgoing Trusted Applications authentication will allow the remote application to trust Crucible. In other words, Crucible will be able to access specified functions and data on the remote application.

To configure Trusted Applications authentication for an outgoing application link:

1. Log in as a system administrator and go to the administration page. Click 'Application Links' in the administration menu. The 'Configure Application Links' page will appear, showing the application links that have been set up.
2. Click the 'Configure' link next to the application link that you want to configure Trusted Applications authentication for.
3. Click the 'Outgoing Authentication' tab. The outgoing authentication page will show, with the 'Trusted Applications' tab displayed.
4. If you are not currently logged into the remote application (or you logged into the remote application under a variant of the application's hostname, e.g. the IP address), a login dialogue will display.
   - Enter the 'Username' and 'Password' for the remote server, (not your local server), and click the 'Login' button. You need to enter the credentials for the remote server, as the remote server needs to be instructed to trust your local server for the Trusted Applications protocol to work. If you are already logged into your remote server, then the appropriate changes can be made without having to log in again.
5. Configure the settings for the Trusted Applications authentication:
   - 'IP Patterns' — Enter the IP addresses (IPv4 only) from which the remote application will accept requests (this effectively is the IP address your local server). You can specify wildcard matches by using an asterisk (*), e.g. '192.111.*.*' (note, you cannot use netmasks to specify network ranges). If you are entering multiple IP addresses, separate them with commas or spaces.

   Please note, if you are setting up Trusted Applications between two applications that both have the Application Links plugin installed, you can leave this field blank (or explicitly use *.+.+.+). However, if your remote application does not have the Application Links plugin installed and you
are configuring the IP Patterns in the remote application (not the Application Links plugin), you must not leave this field blank nor use *.*.*.*. Failure to configure IP address restrictions in this scenario is a security vulnerability, allowing an unknown site to log into your site under a user’s login ID.

Consider the following scenarios, if you want to limit access by using this field:

- If your local application is using a proxy server, you need to add the proxy server’s IP address to this field.
- If your local application is a clustered instance of Confluence, you need to configure the remote server to accept requests from each cluster node. If you do not set up each node appropriately, your Confluence users may not be able to view any information from the remote server. You can set this up by either specifying each individual IP address for each node of the cluster (e.g. 172.16.0.10, 172.16.0.11, 172.16.0.12), or specifying the IP address for the clustered Confluence instance using wildcards (e.g. 172.16.0.*).

- 'URL Patterns' — Enter the URLs in the remote application that your local application will be allowed to access. Each URL corresponds to a particular application function. Enter one URL per line, as follows:
  - If your remote application is JIRA, enter the following URL Patterns: /plugins/servlet/streams,/sr/jira.issueviews:searchrequest,/secure/RunPortlet,/rest,/rpc/soap
  - If your remote application is Confluence, enter the following URL Patterns: /plugins/servlet/stream, /plugins/servlet/applinks/whoami

- 'Certificate Timeout (ms)' — Enter the certificate timeout. The default is 10 seconds. The certificate timeout is used to prevent replay attacks. For example, if a Trusted Applications request is intercepted and (maliciously) re-sent, the application will be able to check when the request was first sent. If the second request is sent more than 10 seconds (or whatever the certificate timeout is set to) after the initial request, it will be rejected. Please note, you should not have to change the default value of this field for most application links. Note that the certificate timeout relies on the clocks on both servers being synchronised.

6. Click the 'Apply' button to save your changes.

**Configuring Trusted Applications for Incoming Authentication**

Configuring incoming Trusted Applications authentication will allow Crucible to trust the remote application. The remote application will be able to access specified functions and data on Crucible.

To configure Trusted Applications authentication for an incoming application link:

1. Log in as a system administrator and go to the administration page. Click 'Application Links' in the administration menu. The 'Configure Application Links' page will appear, showing the application links that have been set up.
2. Click the 'Configure' link next to the application link that you want to configure Trusted Applications authentication for.
3. Click the 'Incoming Authentication' tab. The incoming authentication page will show, with the 'Trusted Applications' tab displayed.
4. The tab will show whether Trusted Applications is currently enabled or not. Use the 'Modify' or 'Configure' button to configure Trusted Applications. The Trusted Applications configuration settings will be displayed:
   - 'IP Patterns' — Enter the IP addresses (IPv4 only) from which our application will accept requests. You can specify wildcard matches by using an asterisk (*), e.g. '192.111.*.*' (note, you cannot use netmasks to specify network ranges). If you are entering multiple IP addresses, separate them with commas or spaces.

   Please note, if you are setting up Trusted Applications between two applications that both have the Application Links plugin installed, you can leave this field blank (or explicitly use *.*.*.*). However, if your remote application does not have the Application Links plugin installed and you are configuring the IP Patterns in the remote application (not the Application Links plugin), you must not leave this field blank nor use *.*.*.*. Failure to configure IP address restrictions in
this scenario is a security vulnerability, allowing an unknown site to log into your site under a user's login ID.

Consider the following scenarios, if you want to limit access by using this field:

- If the remote application is using a proxy server, you need to add the proxy server's IP address to this field.
- If the remote application is a clustered instance of Confluence, you need to accept requests from each cluster node. If you do not specify each node's address, Confluence users may not be able to view any data from your application. You can set this up by either specifying each individual IP address for each node of the cluster (e.g. 172.16.0.10, 172.16.0.11, 172.16.0.12), or specifying the IP address for your clustered Confluence instance using wildcards (e.g. 172.16.0.*).

- 'URL Patterns' — Enter the local URLs that the remote application will be allowed to access. Each URL corresponds to a particular application function. Enter one URL per line, as follows:
  - If your local application is JIRA, enter the following URL Patterns — /plugins/servlet/streams, /sr/jira.issueviews:searchrequest, /secure/RunPortlet, /rest, /rpc/soap
  - If your local application is Confluence, enter the following URL Patterns — /plugins/servlet/streams, /plugins/servlet/applinks/whoami

- 'Certificate Timeout (ms)' — Enter the certificate timeout. The default is 10 seconds. The certificate timeout is used to prevent replay attacks. For example, if a Trusted Applications request is intercepted and (maliciously) re-sent, the application will be able to check when the request was first sent. If the second request is sent more than 10 seconds (or whatever the certificate timeout is set to) after the initial request, it will be rejected. Please note, you should not have to change the default value of this field for most application links. Note that the certificate timeout relies on the clocks on both servers being synchronised.

5. Click the ‘Apply’ button to save your changes.

Related Topics

- Configuring Basic HTTP Authentication for an Application Link
- Configuring OAuth Authentication for an Application Link

Incoming and Outgoing Authentication

When you configure authentication for an application link, you are defining the level of trust between the two linked servers. When configuring a link from one application to another, you can set up:

- **Incoming authentication** (authentication of requests coming from a linked application into this application).
- **Outgoing authentication** (authentication of requests sent from this application to a linked application).

See Configuring Authentication for an Application Link.

Configuring Project Links across Applications

Let's assume that you are managing a project or team. You would like to connect your project's Confluence space with your JIRA project, and link up your team's source repository too.

When you have connected your applications via Application Links, you can also connect the areas of those applications that contain information relating to your project or team. Using **project links** (also called **entity links**) you can associate one or more projects, spaces and repositories across the linked applications.

To connect all the information relating to the project or team that you are managing, you can link one or more of the following:

- JIRA projects.
- Confluence spaces.
Uses for Project Links

The following integration features use project links:

- Activity streams. For example, the project links determine the activity retrieved from JIRA to display in the activity stream of a FishEye repository or a Crucible project.
- The JIRA FishEye plugin. For example:
  - The link between a JIRA project and a FishEye repository determines the repository searched for a particular issue key when displaying the FishEye source tab in JIRA.
  - The link between a JIRA project and a Crucible project determines the Crucible project scanned for review activity when displaying the Crucible reviews tab in JIRA.
  - When you create a defect in Crucible, Crucible will know which JIRA project to put it in.
- Third-party plugins may make use of project links to enrich their functionality too.

Managing Project Links

- Adding Project Links between Applications
- Deleting a Project Link
- Making a Project Link the Primary Link

RELATED TOPICS

Adding an Application Link

Adding Project Links between Applications

Let's assume that you are managing a project or team. You would like to connect your project's Confluence space with your JIRA project, and link up your team's source repository too.

When you have connected your applications via Application Links, you can also connect the areas of those applications that contain information relating to your project or team. Using project links (also called entity links) you can associate one or more projects, spaces and repositories across the linked applications.

To connect all the information relating to the project or team that you are managing, you can link one or more of the following:

- JIRA projects.
- Confluence spaces.
- FishEye repositories.
- FishEye projects. A FishEye 'project' is the Crucible project if you have installed FishEye and Crucible, otherwise it is the paths associated via the 'FishEye Project Content' function in FishEye.
- Crucible projects.
- Bamboo projects.
1. Go to the Crucible administration console and click ‘Projects’.
2. Click ‘Application Links’ in the ‘Operations’ column next to the project that you want to link from. The project links screen will appear.
3. The instructions for adding a project link will vary depending on whether the target application has the Application Links functionality installed:
   
   - If the target application has Application Links:
     a. Click ‘Add Link’. A dropdown menu will appear listing the applications you have already linked to.
     b. In the dropdown menu, click the application that contains the project you want to link to. For example, if you want to link to a specific JIRA project, click the JIRA site that contains that project. If you want to link to a Confluence space, click the Confluence site that contains that space.
     c. Click one of the options on the ‘Authorization required’ screen:
        • ‘Authorize’ — Click this option if you want to grant your project authorised access to the target project. The target application will open in a new window, so that you can log in and authorise access.
        • ‘Skip — your access is anonymous’ — Click this option if you only want to allow anonymous access to the target project.
     d. In the ‘Name or Key’ field, enter the name/key of the project in the remote application that you want to link to. For example, if you want to link to a specific JIRA project, enter the project key. If you want to link to a Confluence space, enter the space key.
     e. Click the ‘Create’ button to create the project link.
   - If the target application does not have Application Links:
     a. Click ‘Add Link’. A dropdown menu will display listing the applications you have already linked to.
     b. In the dropdown menu, click the application that contains the project you want to link to. For example, if you want to link to a specific JIRA project, click the JIRA site that contains that project. If you want to link to a Confluence space, click the Confluence site that contains that space.
     c. In the ‘Key’ field, enter the name/key of the project in the remote application that you want to link to. For example, if you want to link to a JIRA project, enter the project key. If you want to link to a Confluence space, enter the space key.
     d. (optional) Enter the alias for the project in the ‘Alias’ field. This is the display name for the project in your administration console.
     e. Click the ‘Create’ button to create the project link.

Screenshot 1 above: Crucible projects in the administration console

Screenshot 2 above: The project links screen showing the project dropdown list with one project
RELATED TOPICS

Making a Project Link the Primary Link
Deleting a Project Link

Deleting a Project Link

Deleting a project link stops the two projects from sharing information.

If you have set up multiple project links to the same application, for example you have linked a Crucible project to multiple JIRA projects, deleting the primary link will mean that another of the links will be made the primary link.

To delete a project link:

1. Go to the Crucible administration console and click ‘Projects’.
2. Click ‘Application Links’ in the ‘Operations’ column next to the project that you want to link from. The project links screen will appear.
3. Click ‘Delete’ next to the link that you want to delete.
4. A confirmation screen will appear. Click the ‘Confirm’ button to delete the link.

Related Topics

Adding Project Links between Applications
Making a Project Link the Primary Link

Making a Project Link the Primary Link

If you have set up project links to more than one project in the same application, then one of the project links will be marked as the primary link. All outgoing requests will be directed to the primary link.

For example, if you have a Crucible project that is linked to two JIRA projects, you can nominate the link to one of the JIRA projects as the primary link. Every time Crucible requests JIRA information, it will request it from the primary link's JIRA project. Both JIRA projects can still request information from the Crucible project.

To make a project link the primary link:

1. Go to the Crucible administration console and click ‘Projects’.
2. Click ‘Application Links’ in the ‘Operations’ column next to the project that you want to link from. The project links screen will appear.
3. Click the ‘Make Primary’ link in the ‘Action’ column for the project link that you want to make the primary link. A symbol will display in the ‘Primary’ column next to the link.
   Note: The ‘Primary’ column and ‘Make Primary’ link will appear only if you have set up multiple project links to the same application, for example you have linked a Confluence space to a number of JIRA projects.

Related Topics

Adding Project Links between Applications
Deleting a Project Link

Deleting an Application Link

Deleting an application link stops the two applications from sharing information. You will no longer be able to make requests from one application to the other. This means that certain integration features may not work.

If you have set up application links to multiple servers of the same application type, for example you have linked Crucible to multiple JIRA servers, deleting the primary link will mean that another of the links will be made the primary link.
Deleting an application link will also delete all project links set up for that application link.

To delete an application link:

1. Log in as a system administrator and go to the administration page. Click 'Application Links' in the administration menu. The 'Configure Application Links' page will appear, showing the application links that have been set up.
2. Click the 'Delete' link next to the application link that you want to delete. A confirmation screen will be displayed.
3. Click the 'Confirm' button to delete the application link.

RELATED TOPICS

Editing an Application Link
Relocating an Application Link

Editing an Application Link

You can change the details, such as the application name and display URL, for an existing application link.

To edit an application link:

1. Log in as a system administrator and go to the administration page. Click 'Application Links' in the administration menu. The 'Configure Application Links' page will appear, showing the application links that have been set up.
2. Click the 'Configure' link next to the application link that you want to edit the details for. The application details for the application link will be displayed.
3. Update the application details as desired. Please note, you cannot update the Application Type nor the Application URL.
   - 'Application Name' — Update this field to change the display name for the application that you are linking to.
   - 'Display URL' — This URL is used when displaying links to the application in the browser. When creating the application link, you may have used a URL that is not accessible to other users, such as an internal IP address. If so, you can change the display URL to an address in a domain that is accessible to other users.
4. Click the 'Update' button to save your changes.
Making an Application Link the Primary Link

If you have set up application links to more than one of the same application type, for example you have linked Crucible to two JIRA servers, then one of the servers will be marked as the ‘primary’ link. This means that any outgoing requests will be directed to the primary link's application.

For example, if you have linked a Crucible server to two JIRA servers with two-way authentication for both links, you can nominate an application link to one of the JIRA servers as the primary link. Every time Crucible requests JIRA information, it will request it from the primary link's JIRA server. Both JIRA servers can still make requests of the Crucible server.

On this page:

- Making an Application Link the Primary Link
- Notes

Making an Application Link the Primary Link

To make an application link the primary link:

1. Log in as a system administrator and go to the administration page. Click ‘Application Links’ in the administration menu. The ‘Configure Application Links’ page will appear, showing the application links that have been set up.

2. Click the 'Make Primary' link next to the application link that you want to make the primary link. A ‘✔’ symbol will display in the ‘Primary’ column next to the application link.

   The ‘Primary’ column and ‘Make Primary’ link will only display if you have set up application links to
more than one of the same application type, e.g. you have linked your application to two JIRA servers.

Notes

Please read Making a Project Link the Primary Link for information on how primary project links also influence the information shared between servers.

Related Topics

Making a Project Link the Primary Link

Relocating an Application Link

This page describes how to change the location of an application link. You will need to relocate an application link if the target application has moved to a new address.

To relocate an application link:

1. Log in as a system administrator and go to the administration page. Click ‘Application Links’ in the administration menu. The ‘Configure Application Links’ page will appear, showing the application links that have been set up.
2. If the remote application for an application link cannot be reached by your application, the ‘List Application Links’ page will display a warning message (see ‘Relocate Link - Warning Message’ screenshot below).
3. If your remote application has been moved to a different address (rather than just being offline temporarily), click the ‘Relocate’ link in the warning message (see ‘Relocate Link - Updating URL’ screenshot below).
4. Enter the new URL for the remote application of your application link and click ‘Relocate’.
5. You will need to confirm the relocation, if the new URL cannot be contacted. Otherwise, the application link will be updated.

Screenshot above: Relocate link – The warning message

Screenshot above: Relocate link – Updating the URL
Related Topics

Making an Application Link the Primary Link

Upgrading an Application Link

This page describes how to upgrade an existing application link. You may want to upgrade an application link in either of the two situations below:

- Your local Crucible application has been upgraded from a version that does not include Application Links to a version that does.
- Your remote application has been upgraded to a version that includes Application Links. For example, you had set up an application link to a JIRA 4.2 site (does not include Application Links) and have since upgraded to JIRA 4.3 (includes Application Links).

On this page:

- Upgrading an Application Link (Local App Upgraded to Include Application Links)
- Upgrading an Application Link (Remote App Upgraded to Include Application Links)

Upgrading an Application Link (Local App Upgraded to Include Application Links)

When you upgrade from a Crucible version that does not include Application Links to a version that does, you will have the option of converting any Trusted Applications or OAuth links to Application Links. The advantage of converting your links to Application Links is that link configuration will be simplified in future.

To upgrade an application link when Crucible has been upgraded to include Application Links:

1. After your Crucible upgrade, go to the administration console.
2. Click 'Application Links'. The 'Configure Application Links' screen will show the following message: "There are existing Trusted Applications or OAuth relationships that should be upgraded to Application Links. Click here to upgrade."
3. Click the 'Click here to upgrade' link. The 'Existing Trust Relationships' screen will appear, showing all Trusted Applications and OAuth relationships that can be upgraded to Application Links.
4. Click the 'Upgrade to Application Link' link next to the desired trust relationship. The 'Upgrade to Application Link' wizard will appear.
5. Complete the wizard. The process will be similar to adding a new link (described on Adding an Application Link), except that most fields should be pre-filled.

Screenshots above: Upgrading an application link for your local application
Upgrading an Application Link (Remote App Upgraded to Include Application Links)

When an application link is created between a version of Crucible that supports Application Links, and a remote application that does not (either a non-Atlassian product, or an older version of an Atlassian product that did not ship with Application Links), this link is configured to run in 'legacy mode'. While there is no distinguishable difference to a user, connection and configuration without Application Links is a little different. For example:

- Setting up OAuth requires manual configuration by the administrator. If your applications support Application Links, exchange of the consumer keys and public keys is done automatically.
- The Trusted Applications protocol (Atlassian-specific) will not be available for authentication.

If you upgrade your remote application to a version that does support Application Links, the application link will continue to work. Upgrading your link may simplify link configuration and make additional authentication protocols available (as mentioned above).

To upgrade an application link when your remote application has been upgraded to include Application Links:

1. After you have upgraded your remote application to a version that includes Application Links, go to the administration console of your local application. A warning will be displayed, requesting that you upgrade the link to full Application Links mode.
2. Click ‘Upgrade’ in the warning message to start the upgrade wizard. Note the following:
   - You will be prompted to make your application link a reciprocal link. You will need to provide administrator credentials for your remote application, if you choose to do so.
   - If you make your application link a reciprocal link, you will also be able to make reciprocal links for your project links. For example, you may be able to link your JIRA project to a FishEye repository and also make a link from your FishEye repository back to the JIRA project.

Screenshot above: Upgrading an application link for remote application

Configure Application Links

<table>
<thead>
<tr>
<th>Name</th>
<th>Application</th>
<th>Application URL</th>
<th>Configured Authentication</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>REFAPP</td>
<td>Reference Application</td>
<td><a href="http://localhost:5392/refapp">http://localhost:5392/refapp</a></td>
<td>Basic Access</td>
<td>Configure</td>
</tr>
</tbody>
</table>

Screenshot above: Upgrading an application link for remote application
JIRA Integration in Crucible

**JIRA** is Atlassian's issue tracking and project management application. This page describes how to integrate JIRA with Crucible.

**Initial Configuration in JIRA**

Configure the following setting in JIRA:

- **Allow remote API access**

Consider also the following settings, to make full use of the integration between Crucible and JIRA:
Quick Setup using the Crucible Setup Wizard

When you are installing the application, the setup wizard gives you the opportunity to configure the JIRA connection automatically. This is a quick way of setting up your JIRA integration with the most common options. See Configuring JIRA Integration in the Setup Wizard.

If you did not set up the JIRA integration during the Crucible setup, you can configure the integration using the Crucible administration screens as described below.

JIRA Integration using the Crucible Administration Screens

You can further configure the JIRA connections using the Crucible administration screens. There are two parts to the integration process:

- A peer-to-peer link between JIRA and Crucible for sharing information and facilitating integration features. This link is set up using Application Links.
- A client-server link between Crucible and JIRA for delegating user and group management to your JIRA server.

1. Configuring the Application Links with JIRA

Below are step-by-step instructions for creating a two-way trusted applications link between Crucible and JIRA. We assume that your Crucible and JIRA servers are using the same set of users. If not, please refer to Adding an Application Link for information about the various options available.

To add a two-way trusted applications link between Crucible and JIRA:

1. Go to your Crucible administration screen and click Application Links (under ‘Global Settings’).
2. Now:
   a. Click ‘Add Application Link’. Step 1 of the link wizard will appear.
   b. Enter the server URL of the application that you want to link to (the ‘remote application’).
   c. Click the ‘Next’ button.
   d. Enter the following information:
      - ‘Create a link back to this server’ – Tick this check box to add a two-way link between the two applications.
      - ‘Username’ and ‘Password’ – Enter the credentials for a username that has administrator access to the remote application.

Note: These credentials are only used to authenticate you to the remote application, so that Application Links can make the changes required for the new link. The credentials are not saved.
1. Reciprocal Link URL – The URL you give here will override the base URL specified in your remote application's administration console, for the purposes of the application links connection. Application Links will use this URL to access the remote application.

e. Click the 'Next' button.

f. Enter the information required to configure authentication for your application link:

   - 'The servers have the same set of users' – Tick this check box, because the users are the same in both applications.
   - 'These servers fully trust each other' – Tick this check box, because you trust the code in both applications and are sure both applications will maintain the security of their private keys.

For more information about configuring authentication, see Configuring Authentication for an Application Link.

g. Click the 'Create' button to create the application link.

3. If you want to show issues in your Crucible activity streams:

   - Go to the list of application links and click 'JIRA settings' next to the JIRA application link.
   - Tick the 'Include in Activity Streams' check box.
   - Click 'Save'.

4. Connect your projects across the applications:

   - Connect your Crucible projects with your JIRA projects. See the Crucible guide to adding project links.
   - If you are using FishEye, you may want to connect your FishEye repositories with your JIRA projects. See the FishEye guide to adding project links.

2. Connecting to JIRA for User Management

This option is available only with JIRA 4.3 and later.

If you want to manage your Crucible users via JIRA, you need to configure a connection in JIRA and then set up the user directory in Crucible. Both steps are described below.

2.1. To configure a new connection for user management in JIRA:

   1. Log in to JIRA as a user with the 'JIRA System Administrators' global permission.
   2. Click the 'Administration' link on the JIRA top navigation bar.
      a. Go to the JIRA administration screen for configuring the applications that have been set up to use JIRA for user management:
         - In JIRA 4.3: Click 'Other Applications' in the 'Users, Groups & Roles' section of the JIRA administration screen.
         - In JIRA 4.4: Select 'Administration' > 'Users' > 'JIRA User Server'.
   b. Add an application.
   c. Enter the application name and password that your application will use when accessing JIRA.
   d. Enter the IP address or addresses of your application. Valid values are:
      - A full IP address, e.g. 192.168.10.12.
      - A wildcard IP range, using CIDR notation, e.g. 192.168.10.1/16. For more information, see the introduction to CIDR notation on Wikipedia and RFC 4632.
   e. Save the new application.

2.2. To set up the JIRA user directory in Crucible:

Go to your Crucible administration screen.

   1. Click Authentication (under 'Security Settings').
   2. Click Setup JIRA/Crowd authentication. Note, if LDAP authentication has already been set up, you will need to remove that before connecting to JIRA for user management.
   3. Make the following settings:

   | Authenticate against | Select a JIRA instance |
### Application name and password
Enter the values that you defined for your application in the settings on JIRA.

### JIRA URL
The web address of your JIRA server. Examples:

```
http://www.example.com:8080/jira/
http://jira.example.com
```

### Auto-add
Select Create a FishEye user on successful login so that your JIRA users will be automatically added as a FishEye user when they first log in.

### Periodically synchronise users with JIRA
Select Yes to ensure that JIRA will synchronize all changes in the user information on a regular basis. Change the value for Synchronise Period if required.

### When Synchronisation Happens
Select an option depending on whether you want to allow changes to user attributes from within FishEye.

### Single Sign On
Select Disabled. SSO is not available when using JIRA for user management and if enabled will make the integration fail.

4. Click **Next** and select at least one user group to be synchronised from JIRA. If necessary, you could create a new group in JIRA, such as 'fisheye-users', and select this group here.

5. Click **Save**.

### Obtaining Subtask Values for Crucible Configuration

The subtask ID is required (along with the subtask resolution ID and the subtask resolution action ID) to enable the creation of issues from a Crucible comment. This is the subtask type that will be created when you create a JIRA subtask in Crucible.

To set this up in Crucible:

1. Enable subtasks on your JIRA site. See the [JIRA documentation](https://confluence.atlassian.com/display/DOCS/Introduction+to+Using+JIRA+Subtasks+and+Epics+in+Crucible).
2. Return to the Crucible administration screen and click 'Application Links' under the 'Global Settings' menu in the left-hand navigation bar.
3. Click '**JIRA settings**' next to the application link for the required JIRA server. The 'Update JIRA Server' screen will appear.
4. Click '**Test**'. The field for 'Subtask Type ID' will change to a dropdown menu, showing the available subtask types. Choose the correct one. The field for 'Subtask Resolution' will also turn into a dropdown menu. Select the required item from this menu as well.
   - **Note that you will need to select a subtask type that does not have a required field** (due to CRU C-2667). Our suggested workarounds at this time are:
     - Set a default value for the required fields, or
     - Create a separate issue type, eg "Crucible Subtask" that has a workflow that suits Crucible (e.g. Open -> Close with no required fields). This avoids changing existing workflows, and can be more obvious in JIRA that it has a looser workflow.
5. Click '**Save**'.
6. Open your JIRA site and go to 'Administration' > 'Workflows'. The 'Workflows' screen appears. By default, the 'JIRA' workflow is shown on screen in a table.
7. Click the 'Steps' link in the far right table cell. The 'View Workflow Steps – JIRA' page appears.
8. The 'Subtask Resolution Action ID' is in the 'Open' row, under the 'Transitions' column. Look at the link in that cell named 'Resolve Issue'. The ID number is shown in brackets next to that heading 'Resolve Issue' (shown in the screenshot below as 5).

9. Enter the number into the field in Crucible.

10. Save your Crucible configuration settings.

Screenshot above: Choosing subtask values

---

Update JIRA Server

Server Details
Name: jacc
Authentication

- [ ] Use Basic Authentication

Username: crucible-user
Password: *********

Options

- Include in Activity Streams
- Allow Time Tracking submission from reviews

Subtask Settings

Leave these fields blank to disable subtasks.

Subtask Type ID: 17
Subtask Resolution Action ID: 2
Subtask Resolution ID: 9

Allow Unassigned: [ ] Yes [ ] No

Pressing Test will retrieve subtask types and resolutions from JIRA.

Test Save Cancel

Screenshot: Obtaining the subtask resolution action ID
User Management Limitations and Recommendations

This page describes the optimal configurations and limitations that apply when you are connecting Crucible to JIRA for user management.

On this page:

- Recommendations for Connecting to JIRA for User Management
  - Single Sign-On Across Multiple Applications is Not Supported
  - Custom Application Connectors are Not Supported
  - Custom Directories are Not Supported
  - Optimal Number of Users and Applications
  - Recommendations

Recommendations for Connecting to JIRA for User Management

Please consider the following limitations and recommendations when connecting to a JIRA server for user management.

Single Sign-On Across Multiple Applications is Not Supported

When you connect to JIRA for user management, you will not have single sign-on across the applications connected in this way. JIRA, when acting as a directory manager, does not support SSO.

Custom Application Connectors are Not Supported

JIRA, Confluence, FishEye, Crucible and Bamboo can connect to a JIRA server for user management. Custom application connectors will need to use the new REST API.
**Custom Directories are Not Supported**

Earlier versions of JIRA supported OSUser Providers. It was therefore possible write a special provider to obtain user information from any external user directory. This is no longer the case.

**Optimal Number of Users and Applications**

Please consider the following limitations when connecting to a JIRA server for user management:

- Maximum 500 users.
- Maximum 5 connected applications.

**Recommendations**

<table>
<thead>
<tr>
<th>Your environment</th>
<th>Recommendation</th>
</tr>
</thead>
<tbody>
<tr>
<td>If <strong>all</strong> the following are true:</td>
<td>Your environment meets the optimal requirements for using JIRA for user management.</td>
</tr>
<tr>
<td>- You have fewer than 500 users.</td>
<td></td>
</tr>
<tr>
<td>- You want to share user and group management across just a few applications, such as one JIRA server and one Confluence server, or two JIRA servers.</td>
<td></td>
</tr>
<tr>
<td>- You do not need single sign-on (SSO) between JIRA and Confluence, or between two JIRA servers.</td>
<td></td>
</tr>
<tr>
<td>- You do not have custom application connectors. Or, if you do have them, you are happy to convert them to use the new REST API.</td>
<td></td>
</tr>
<tr>
<td>- You are happy to shut down all your servers when you need to upgrade JIRA.</td>
<td></td>
</tr>
<tr>
<td>If <strong>one or more</strong> of the following are true:</td>
<td>We recommend that you install <a href="https://www.atlassian.com/products/crowd">Atlassian Crowd</a> for user management and SSO.</td>
</tr>
<tr>
<td>- You have more than 500 users.</td>
<td></td>
</tr>
<tr>
<td>- You want to share user and group management across more than 5 applications.</td>
<td></td>
</tr>
<tr>
<td>- You need single sign-on (SSO) across multiple applications.</td>
<td></td>
</tr>
<tr>
<td>- You have custom applications integrated via the Crowd SOAP API, and you cannot convert them to use the new REST API.</td>
<td></td>
</tr>
<tr>
<td>- You are not happy to shut down all your servers when you need to upgrade JIRA.</td>
<td></td>
</tr>
</tbody>
</table>
If you are considering creating a custom directory connector to define your own storage for users and groups...

<table>
<thead>
<tr>
<th>Please see if one of the following solutions will work for you:</th>
</tr>
</thead>
<tbody>
<tr>
<td>• If you have written a custom provider to support a specific LDAP schema, please check the supported LDAP schemas to see if you can use one of them instead.</td>
</tr>
<tr>
<td>• If you have written a custom provider to support nested groups, please consider enabling nested groups in the supported directory connectors instead.</td>
</tr>
<tr>
<td>• If you have written a custom provider to connect to your own database, please consider loading the data into the application's database instead.</td>
</tr>
<tr>
<td>• If you need to keep the custom directory connection, please consider whether Atlassian Crowd meets your requirements. See the documentation on Creating a Custom Directory Connector.</td>
</tr>
</tbody>
</table>

**Related Topics**

- JIRA Integration in Crucible

---

**Trusted Applications**

A 'trusted application' is an application that Crucible will allow to access specific functions in Crucible on behalf of any user, without the user logging in to Crucible. You can configure Trusted Applications authentication between FishEye, Crucible, JIRA (3.12 and later) and Confluence (2.7 and later).

Using the bundled Application Links plugin, you can configure application links with Trusted Applications authentication as well as other types of authentication.

For further instructions, please see Configuring Application Links.

**Related Topics**

- Configuring Application Links
- JIRA Integration in Crucible

---

**Crucible Release Notes**

⚠️ Crucible 2.8 has now been released. Read the Release Notes.

**Crucible Release Notes and Changelogs**

- **Security advisories**
  - FishEye and Crucible Security Advisory 2012-08-21
  - FishEye and Crucible Security Advisory 2012-05-17
  - FishEye and Crucible Security Advisory 2012-01-31
  - FishEye and Crucible Security Advisory 2011-11-22
  - FishEye and Crucible Security Advisory 2011-05-16
  - FishEye and Crucible Security Advisory 2011-01-12
  - Crucible Security Advisory 2010-06-16
  - Crucible Security Advisory 2010-05-04
- Crucible Upgrade Guide
• Upgrading to a new version of Crucible
• Upgrading from FishEye to Crucible
• Crucible Release Summary
• Crucible 2.8 Release Notes
  • Crucible 2.8 Upgrade Guide
  • Crucible 2.8 Changelog
• Crucible 2.7 Release Notes
  • Crucible 2.7 Upgrade Guide
  • Crucible 2.7 Changelog
• Crucible 2.6 Release Notes
  • Crucible 2.6 Upgrade Guide
  • Crucible 2.6 Changelog
• Crucible 2.5 Release Notes
  • Crucible 2.5 Upgrade Guide
  • Crucible 2.5 Changelog
• Crucible 2.4 Release Notes
  • Crucible 2.4 Upgrade Guide
  • Crucible 2.4 Changelog
• Crucible 2.3 Release Notes
  • Crucible 2.3 Upgrade Guide
  • Crucible 2.3 Changelog
• Crucible 2.2 Release Notes
  • Crucible 2.2 Upgrade Guide
  • Crucible 2.2 Changelog
• Crucible 2.1 Release Notes
  • Crucible 2.1 Upgrade Guide
  • Crucible 2.1 Changelog
• Crucible 2.0 Release Notes
  • Crucible 2.0 Upgrade Guide
  • Crucible 2.0 Changelog
• Crucible 2.0 Beta Release Notes
  • JIRA Integration in Crucible 2.0 Beta
  • Crucible 2.0 Beta Upgrade Notes
  • Crucible 2.0 Beta Reviewer’s Guide
• Crucible 1.6 Release Notes
  • Crucible 1.6 Upgrade Guide
  • Crucible 1.6 Changelog
• Crucible 1.5 Release Notes
  • Crucible 1.5 Upgrade Guide
  • Crucible 1.5 Changelog
• Crucible 1.2 Release Notes
  • Crucible 1.2 Upgrade Guide
  • Crucible 1.2 Changelog
• Crucible 1.1 Release Notes
  • Crucible 1.1 Upgrade Guide
  • Crucible 1.1 Changelog

• For changes prior to 1.1, see the 1.0.x Changelog

Installation

You can download Crucible from here.

If upgrading from a previous version, please follow the Upgrade Guide.

• As of version 1.0, Crucible now requires a JVM version 1.5 or later. Previously, 1.4+ was required.
• Crucible 1.1.4 includes FishEye 1.3.8.
Documentation for Crucible 2.8

Upgrading from 1.0.4 (or earlier) will force a complete re-index of P4 repositories.

Crucible Upgrade Guide
Upgrading to a new version of Crucible
Upgrading from FishEye to Crucible

Upgrading to a new version of Crucible
This page describes the recommended method of upgrading to a new version of Crucible.
Read about how your Crucible installation works with FishEye.

Before you start
Back up your entire Crucible instance (see Backing up and restoring Crucible data), i.e.
If you are backing up your Crucible instance via the Admin interface, tick all of the ' Include'
checkboxes (e.g. plugins, templates, uploads, SQL database, etc).
If you are backing up your Crucible instance using the command-line interface, do not use any excl
usion options.
Read the Release Notes and Changelog and version-specific Upgrade Guides for the version you are
upgrading to, as well as any versions you are skipping.
Check the Supported Platforms to ensure that your system meets the requirements for the new version.
Download the Crucible zip file.
On this page:
Before you start
Upgrade procedure
Method 1: Using a FISHEYE_INST directory
Method 2: Without a FISHEYE_INST directory
Method 3: Without a FISHEYE_INST directory, but would like to set one up
Checking for known issues and troubleshooting the Crucible upgrade

Upgrade procedure
Your upgrade procedure depends on whether you are using a FISHEYE_INST directory (i.e. "FishEye instance"
directory).
The FISHEYE_INST directory is the FishEye data directory (not the installation directory) and has a
location defined by the FISHEYE_INST environment variable. It is used to keep the FishEye data
completely separate from the FishEye/Crucible application files. We recommend that you configure
FishEye/Crucible to use a FISHEYE_INST directory for production instances. Read more about FISHEYE
_INST in Installing FishEye on Windows or Installing FishEye on Linux and Mac.
The <FishEye home directory> is the location of the FishEye/Crucible application files.
NOTE: For all methods below: if you previously ran FishEye as a Windows service and are installing the new
version of FishEye in a new location, you need to reinstall FishEye as a Windows service. Make sure to run
...\wrapper\bin\Fisheye-Uninstall-NTService.bat from the old location. Subsequently, prior to starting the new
instance of FishEye, follow these instructions to set up Fisheye as a Windows service again. Make sure to copy
over the changes from the old ...\wrapper\conf\wrapper.conf file to the new wrapper.conf file.
Method 1: Using a FISHEYE_INST directory

If you have FishEye/Crucible configured to use a FISHEYE_INST directory, then follow the instructions below.
This is the recommended scenario for production installations.
1. Shut down your existing FishEye/Crucible server.
2. Make a backup of your FISHEYE_INST directory.
3.
Created by Atlassian in 2012. Licensed under a Creative Commons Attribution 2.5 Australia License.

232


Method 2: Without a FISHEYE_INST directory

If you do not have FishEye/Crucible configured to use a FISHEYE_INST directory and do not want to set one up, then follow the instructions below. The <FishEye home directory> is the location of the existing FishEye/Crucible installation. Note that this is the typical scenario for evaluation installations, and is not recommended for production installations.

You will need to copy some files from your old FishEye/Crucible installation to your new one.

1. Download FishEye or Crucible.
2. Extract the new FishEye/Crucible archive into a directory such as <New FishEye home directory>.
3. Shut down the old FishEye/Crucible instance if it is running.
4. Copy <FishEye home directory>/config.xml to <New FishEye home directory>.
5. Delete the <New FishEye home directory>/var and <New FishEye home directory>/cache directories.
6. Copy (or move) the <FishEye home directory>/var directory to <New FishEye home directory>/var.
7. Copy (or move) the <FishEye home directory>/cache directory to <New FishEye home directory>/cache.
8. Start FishEye/Crucible from the new installation by running <New FishEye home directory>/bin/run.sh. (Use run.bat on Windows.)

Method 3: Without a FISHEYE_INST directory, but would like to set one up

If you do not have FishEye/Crucible configured to use a FISHEYE_INST directory but would like to set one up, then follow the instructions below. You may wish to do this when reconfiguring an existing installation for a production environment.

The FISHEYE_INST directory is the FishEye data directory, which has a location defined by the FISHEYE_INST environment variable, and which should be completely separate from the <FishEye home directory>. The <FishEye home directory> is the location of the existing FishEye/Crucible installation.

1. Download FishEye or Crucible.
2. Shut down the existing FishEye/Crucible instance if it is running.
3. Set up the FISHEYE_INST environment variable, then create the FISHEYE_INST directory on your file system.
4. Copy <FishEye home directory>/config.xml to the FISHEYE_INST directory.
5. Copy the <FishEye home directory>/var directory to the FISHEYE_INST directory.
6. Copy the <FishEye home directory>/cache directory to the FISHEYE_INST directory.
7. Extract the new FishEye/Crucible archive into a directory such as <New FishEye home directory>.
8. Start FishEye/Crucible from the new installation by running <New FishEye home directory>/bin/run.sh. (Use run.bat on Windows.)
   - If your configuration is not automatically picked up and you cannot see your existing repositories, check your Administration > Sys-Info page, where you will see information about the <FishEye home directory> and FISHEYE_INST. Check that your FISHEYE_INST is pointing to the right directory.
Checking for known issues and troubleshooting the Crucible upgrade

If something is not working correctly after you have completed the steps above to upgrade your Crucible installation, please check for known Crucible issues and try troubleshooting your upgrade as described below:

- **Check for known issues.** Sometimes we find out about a problem with the latest version of Crucible after we have released the software. In such cases we publish information about the known issues in the Crucible Knowledge Base. Please check the [Fisheye and Crucible Known Issues](https://confluence.atlassian.com/display/FISHEYE/known%20issues) in the Crucible Knowledge Base and follow the instructions to apply any necessary patches if necessary.

- **Did you encounter a problem during the Crucible upgrade?** Please refer to the guide to [troubleshooting upgrades](https://confluence.atlassian.com/display/FISHEYE/troubleshooting) in the Crucible Knowledge Base.

- **If you encounter a problem during the upgrade and cannot solve it**, please create a [support ticket](https://confluence.atlassian.com/display/FISHEYE/support%20ticket) and one of our support engineers will help you.

Upgrading from FishEye to Crucible

If you have been using FishEye and now want to move to Crucible, you can do this without losing your FishEye repositories.

Read about how your Crucible installation works with FishEye.

**Before you start**

We strongly recommend you make a backup of your data before following the steps below. Refer to the documentation on [making a backup](https://confluence.atlassian.com/display/FISHEYE/making%20a%20backup).

**On this page:**

- Before you start
- Upgrade procedure
  - Method 1: Without a FISHEYE_INST Directory (default)
  - Method 2: Using a FISHEYE_INST Directory
  - Method 3: Without a FISHEYE_INST Directory, but intending to set one up
- Initial Crucible configuration

**Upgrade procedure**

- Follow **Method 1** if you have a default configuration and are not using a FISHEYE_INST directory (that is, your FishEye binaries and data are all stored under the same location, in the default `<FishEye home directory>`). This is the typical scenario for evaluation installations.

- Follow **Method 2** below if you have FishEye configured to use a FISHEYE_INST directory (that is, your FishEye binaries are stored in the `<FishEye home directory>`, separate from your FishEye data directory with a location defined by FISHEYE_INST). This is the recommended scenario for production installations.

- Follow **Method 3** if you are not using a FISHEYE_INST directory but would now like to start using one. You may wish to do this when configuring an existing installation for a production environment.

Read more about the FISHEYE_INST environment variable.

**Method 1: Without a FISHEYE_INST Directory (default)**

1. **Download** Crucible.
2. Extract the new Crucible archive into a directory such as `<New FishEye directory>`.
3. Shut down the old FishEye instance if it is running.
4. Copy `<FishEye home directory>/config.xml` to `<New FishEye directory>`.
5. Delete the `<New FishEye directory>/var` directory.
6. Copy the `<FishEye home directory>/var` directory to `<New FishEye directory>/var`.
7. Follow any version-specific instructions found in the Release Notes.
8. Start Crucible from the new installation by running `<New FishEye directory>/bin/run.sh` (Use `run.bat` on Windows).
9. Follow the initial configuration steps outlined below.

Method 2: Using a FISHEYE_INST Directory
1. Shutdown your existing fishEye server.
2. Make a backup of your FISHEYE_INST directory.
3. Download Crucible and unzip the archive into a folder. This document assumes you have extracted your Crucible zip file into a directory called `<New FishEye directory>`.
4. Leave your FISHEYE_INST environment variable set to its existing location. Crucible will use this variable.
5. Start Crucible from the new installation by running `<New FishEye directory>/bin/run.sh` (Use `run.bat` on Windows).
6. Follow the initial configuration steps outlined below.

Method 3: Without a FISHEYE_INST Directory, but intending to set one up
1. Shut down the old FishEye instance if it is running.
2. Set up the FISHEYE_INST environment variable, then create the FISHEYE_INST directory on your filesystem.
3. Copy the `<FishEye home directory>/config.xml` to the FISHEYE_INST directory.
4. Copy the `<FishEye home directory>/var` directory to the FISHEYE_INST directory.
5. Download Crucible.
6. Extract the new Crucible archive into a directory such as `<New FishEye directory>`.
7. Start Crucible from the new installation by running `<New FishEye directory>/bin/run.sh` (Use `run.bat` on Windows).
8. Follow the initial configuration steps outlined below.
9. If you configuration is not automatically picked up and you cannot see your existing repositories, check your Administration > Sys-Info page, where you will see information about the <New FishEye directory> and FISHEYE_INST. Check that FISHEYE_INST is pointing to the correct directory.

**Initial Crucible configuration**
2. The first time you run FishEye, enter your Crucible license key. To do this, update your Crucible license by opening ‘Administration’, then ‘Sys-Info/Support’. On this screen, you can enter your Crucible license key. You can view your license key [here](#). The Crucible functionality will be instantly unlocked.
3. If you do not already have user accounts configured, you will need to do this via the Administration screens or by configuring Crucible/FishEye to use external authentication. To add users:
   - Open the FishEye Administration screens at [http://HOSTNAME:8060/admin/](http://HOSTNAME:8060/admin/).
   - Click ‘Users/Security’ under ‘Global Settings’ in the ‘Admin Menu’.
   - Read more details about the different ways of creating users.
4. Crucible can email each review participant on a range of changes. Each user can then set up their own preferences. This is described in the User Profile guide. First, you must set up the SMTP Server.

**Crucible 2.8 Release Notes**

15 August 2012

With great pleasure, Atlassian presents Crucible 2.8, introducing social features and performance
improvements.

See the [change log](#) for Crucible 2.8.x minor releases.

Visit our issue tracker to see the full list of improvements and bug fixes in Crucible and FishEye for this release.

- **Upgrading from a previous version of Crucible.** Upgrading Crucible should be fairly straight forward. *We strongly recommend that you back up Crucible before upgrading.* Please refer to the [Crucible 2.8 Upgrade Guide](#) for further essential information about your upgrade.

- **Known Issues.** Please check the [important technical advisories](#) on the front page of the Knowledge Base for information about any known issues for this release.

Highlights of this release:

- [Mentions](#)
- [Shares](#)
- [Improved performance for the projects listing](#)
- [Support for Subversion 1.7](#)
- [End of life announcements](#)

Providing feedback:

*Please log your votes and issues.* They help us decide what needs doing, and are much appreciated!

---

1

**Mentions**

In Crucible 2.8 you can use mentions to notify other users in the changeset discussions, review comments and snippet comments. Simply type @ and the name of the person you would like to notify to create a mention.
You can now share reviews with other users of Crucible as well as external people via their email. Sharing Crucible content has never been easier.
Improved performance for the projects listing

We improved the rendering performance of the projects listing in Crucible.

Support for Subversion 1.7

Crucible now supports Subversion 1.7.

End of life announcements

As announced earlier, support for IBM ClearCase repositories has been removed in Crucible 2.8.

The Crucible 2.8 team

Development

Core team

Geoff Crain
Tom Davies
Brendan Humphreys
Conor MacNeill
Richard Stephens

Team lead

Nick Pellow
Crucible 2.8 Changelog

This page contains information about the Crucible 2.8 minor releases. FishEye license holders should also check the FishEye 2.8 Changelog. See the Crucible 2.8 Release Notes for details of what's new in 2.8.0.

⚠️ Please read the Crucible 2.8 Upgrade Guide before upgrading to any of the minor releases below.

On this page:

- From 2.8.0 to 2.8.1
- Crucible 2.8

From 2.8.0 to 2.8.1

29 August 2012

This is a bug fix release. The complete list of issues is below.
## JIRA Issues (2 issues)

<table>
<thead>
<tr>
<th>Type</th>
<th>Key</th>
<th>Summary</th>
<th>Assignee</th>
<th>Reporter</th>
<th>Priority</th>
<th>Status</th>
<th>Resolution</th>
<th>Created</th>
<th>Updated</th>
</tr>
</thead>
</table>

---

### Crucible 2.8

**15 August 2012**

The complete list of issues is below.

## JIRA Issues (16 issues)

<table>
<thead>
<tr>
<th>Type</th>
<th>Key</th>
<th>Summary</th>
<th>Assignee</th>
<th>Reporter</th>
<th>Priority</th>
<th>Status</th>
<th>Resolution</th>
<th>Created</th>
<th>Updated</th>
</tr>
</thead>
<tbody>
<tr>
<td>📝</td>
<td>CRUC-6 156</td>
<td>Test upgrade from 2.7 to 2.8 on all Supported DBs</td>
<td>Unassigned</td>
<td>Nick Pellow [Atlassian]</td>
<td>⚠️</td>
<td>Close d</td>
<td>Fixed</td>
<td>Jun 20, 2012</td>
<td>Jul 01, 2012</td>
</tr>
<tr>
<td>📝</td>
<td>CRUC-6 104</td>
<td>Allow sharing reviews using emails or account</td>
<td>Unassigned</td>
<td>Felipe Kraemer [Atlassian]</td>
<td>⚠️</td>
<td>Close d</td>
<td>Fixed</td>
<td>Apr 04, 2012</td>
<td>Aug 13, 2012</td>
</tr>
<tr>
<td>CRUC-6</td>
<td>Issue Title</td>
<td>Description</td>
<td>Author</td>
<td>Assigned To</td>
<td>Status</td>
<td>Fixed</td>
<td>Time</td>
<td></td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
<td>-------------</td>
<td>--------</td>
<td>-------------</td>
<td>--------</td>
<td>-------</td>
<td>------</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>time of a repository</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Crucible 2.8 Upgrade Guide

Below are some important notes on upgrading to Crucible 2.8. For details of the new features and improvements in this release, please read the Crucible 2.8 Release Notes.

On this page:

- Upgrade Notes
- Upgrade Procedure
- Checking for Known Issues and Troubleshooting the Crucible Upgrade

Related pages:

- Crucible 2.8 Release Notes

Upgrade Notes

Crucible 2.8

- A minor schema change has been made to the Crucible SQL Schema in 2.8. If you need to rollback to an earlier version, you will need to restore from a backup to do so. Be sure to take a backup before you upgrade.
  
  Please also be sure to read the End of Life Announcements for FishEye/Crucible 2.8.

Upgrade Procedure

⚠️ Before you begin

- Test your upgrades in your test environment before rolling into production.
- Back up your entire Crucible instance (see Backing up and restoring Crucible data), i.e.
  - If you are backing up your Crucible instance via the Admin interface, tick all of the 'Include' checkboxes (e.g. plugins, templates, uploads, SQL database, etc).
  - If you are backing up your Crucible instance using the command-line interface, do not use any exclusion options.

If you are already running a version of Crucible, please follow the instructions on Upgrading to a new version of...
Checking for Known Issues and Troubleshooting the Crucible Upgrade

If something is not working correctly after you have completed the steps above to upgrade your Crucible installation, please check for known Crucible issues and try troubleshooting your upgrade as described below:

- **Check for known issues.** Sometimes Atlassian finds out about a problem with the latest version of Crucible after the software is released. In such cases we publish information about the known issues in the Crucible Knowledge Base. Please check the [Crucible 2.8 Known Issues](#) in the Crucible Knowledge Base and follow the instructions to apply any necessary patches if necessary.

- **Did you encounter a problem during the Crucible upgrade?** Please refer to the guide to [troubleshooting upgrades](#) in the Crucible Knowledge Base.

- If you encounter a problem during the upgrade and cannot solve it, please create a [support ticket](#) and one of our support engineers will help you.

Crucible 2.7 Release Notes

7 September 2011

With great pleasure, Atlassian presents Crucible 2.7 featuring new JIRA Transitions and Review Reminders.

Highlights of this Release:

- [JIRA Transitions in Crucible](#)
- [Review Reminders](#)
- [Small Improvements](#)

Highlights of Crucible 2.7

1

**JIRA Transitions in Crucible**

For Crucible reviews that are linked to JIRA issues, you can now transition these JIRA issues through workflow from within Crucible.

After **closing a review**, Crucible will now display any available transitions for the linked JIRA issue.
Review Reminders

Crucible will now 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.
Small Improvements

Crucible 2.7 also comes bundled with numerous other bug fixes and improvements, including:

- Syntax highlighting for Java 7, Groovy, Velocity and Scala
- Crucible can now run on Java 7
- Improved user interface for the administration screens
- Improved plugin points for developers

Visit our issue tracker to see the full list of improvements and bug fixes in Crucible and FishEye for this release.

Release Notices

- **Upgrading from a previous version of Crucible.** Upgrading Crucible should be fairly straight forward. *We strongly recommend that you back up Crucible before upgrading.* Please refer to the [Crucible 2.7 Upgrade Guide](#) for further essential information about your upgrade.

- **Known Issues.** Please check the [important technical advisories](#) on the front page of the Knowledge Base for information about any known issues for this release.

Crucible 2.7 Changelog

This page contains information about the Crucible 2.7 minor releases. FishEye license holders should also check the [FishEye 2.7 Changelog](#).
See the Crucible 2.7 Release Notes for details of what's new in 2.7.0.

⚠️ Please read the Crucible 2.7 Upgrade Guide before upgrading to any of the minor releases below.

On this page:

- From 2.7.14 to 2.7.15
- From 2.7.13 to 2.7.14
- From 2.7.12 to 2.7.13
- From 2.7.11 to 2.7.12
- From 2.7.10 to 2.7.11
- From 2.7.9 to 2.7.10
- From 2.7.8 to 2.7.9
- From 2.7.7 to 2.7.8
- From 2.7.6 to 2.7.7
- From 2.7.5 to 2.7.6
- From 2.7.4 to 2.7.5
- From 2.7.3 to 2.7.4
- From 2.7.2 to 2.7.3
- From 2.7.1 to 2.7.2
- From 2.7.0 to 2.7.1

From 2.7.14 to 2.7.15

10 July 2012

This is a bug fix release. The complete list of issues is below.

<table>
<thead>
<tr>
<th>JIRA Issues (8 issues)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Type</strong></td>
</tr>
<tr>
<td>#</td>
</tr>
<tr>
<td>-----</td>
</tr>
<tr>
<td>FE-4178</td>
</tr>
</tbody>
</table>
From 2.7.13 to 2.7.14

12 June 2012

This is a bug fix release. The complete list of issues is below.

<table>
<thead>
<tr>
<th>JIRA Issues (6 issues)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Type</strong></td>
</tr>
<tr>
<td><img src="image1" alt="Image" /></td>
</tr>
</tbody>
</table>
that interact with JIRA will not work

<table>
<thead>
<tr>
<th>#</th>
<th>Key</th>
<th>Summary</th>
<th>Assignee</th>
<th>Reporter</th>
<th>Priority</th>
<th>Status</th>
<th>Resolution</th>
<th>Created</th>
<th>Updated</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>FE-4106</td>
<td>Fisheye 2.7.13 startup error when FISHEYE_INST have spaces</td>
<td>Nick Pellow [Atlassian]</td>
<td>Leo Leung</td>
<td></td>
<td>Close</td>
<td>Fixed</td>
<td>May 24, 2012</td>
<td>Jun 08, 2012</td>
</tr>
</tbody>
</table>

From 2.7.12 to 2.7.13

21 May 2012

This is a bug fix release. The complete list of issues is below.

**JIRA Issues** (39 issues)

<table>
<thead>
<tr>
<th>Type</th>
<th>Key</th>
<th>Summary</th>
<th>Assignee</th>
<th>Reporter</th>
<th>Priority</th>
<th>Status</th>
<th>Resolution</th>
<th>Created</th>
<th>Updated</th>
</tr>
</thead>
<tbody>
<tr>
<td>CRUC-6092</td>
<td>FE-4019</td>
<td>FE-4018</td>
<td>FE-4017</td>
<td>FE-3989</td>
<td>FE-3922</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>----------</td>
<td>---------</td>
<td>---------</td>
<td>---------</td>
<td>---------</td>
<td>---------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Email generated via “Tools -&gt; Email Review” has incorrect comment URLs</strong></td>
<td><strong>web-server/max-th reads increases from 20 to 150</strong></td>
<td><strong>EyeQL Search Results not displaying author fields</strong></td>
<td><strong>Add authenticated user in http response headers for logging ability in reverse proxy</strong></td>
<td><strong>Allow disabling of precise contentHash to csid mapping</strong></td>
<td><strong>Add custom</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Resolved</td>
<td>Fixed</td>
<td>Fixed</td>
<td>Fixed</td>
<td>Fixed</td>
<td>Fixed</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>#</td>
<td>Issue Key</td>
<td>Summary</td>
<td>Assignees</td>
<td>Status</td>
<td>Resolution</td>
<td>Created</td>
<td>Modified</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----</td>
<td>----------------</td>
<td>-------------------------------------------------------------------------</td>
<td>----------------------------------------</td>
<td>---------</td>
<td>------------</td>
<td>---------</td>
<td>-----------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Issue</td>
<td>Title</td>
<td>Description</td>
<td>Reporter</td>
<td>Status</td>
<td>Fixed</td>
<td>Created</td>
<td>Updated</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>-------</td>
<td>-------------</td>
<td>----------</td>
<td>--------</td>
<td>-------</td>
<td>---------</td>
<td>---------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FE-4060</td>
<td>CC Indexing</td>
<td>Can fetch version 0 and index it even if before the start date.</td>
<td>Conor MacNeill [Atlassian]</td>
<td>Closed</td>
<td>Fixed</td>
<td>Apr 17, 2012</td>
<td>Apr 18, 2012</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FE-4043</td>
<td>Handle NPE caused by changes et without a position value in Commit Graph</td>
<td></td>
<td>Conor MacNeill [Atlassian]</td>
<td>Closed</td>
<td>Fixed</td>
<td>Apr 10, 2012</td>
<td>Apr 15, 2012</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FE-4036</td>
<td>P4 GetLatestRevision will fail for paths with no revisions - leads to excessive logging.</td>
<td></td>
<td>Conor MacNeill [Atlassian]</td>
<td>Closed</td>
<td>Fixed</td>
<td>Apr 05, 2012</td>
<td>Apr 10, 2012</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FE-4034</td>
<td>Copy from a file which existed</td>
<td></td>
<td>Conor MacNeill [Atlassian]</td>
<td>Closed</td>
<td>Fixed</td>
<td>Apr 03, 2012</td>
<td>Apr 10, 2012</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ID</td>
<td>Title</td>
<td>Description</td>
<td>Author</td>
<td>Assignee</td>
<td>Resolution</td>
<td>Created</td>
<td>Updated</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>----------------------------------------------------------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>-------------------</td>
<td>-------------------</td>
<td>------------</td>
<td>-----------</td>
<td>-----------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Issue</td>
<td>Description</td>
<td>Assignee</td>
<td>Reporter</td>
<td>Status</td>
<td>Fixed</td>
<td>Date</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>-------------</td>
<td>----------</td>
<td>----------</td>
<td>--------</td>
<td>-------</td>
<td>------</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FE-3984</td>
<td>JIRA user management fails if there are no groups selected to sync</td>
<td>Brendan Humphreys [Atlassian]</td>
<td>Ajay Sridhar [Atlassian]</td>
<td>Closed</td>
<td>Fixed</td>
<td>Mar 06, 2012</td>
<td>Apr 04, 2012</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ticket</td>
<td>Summary</td>
<td>Assigned To</td>
<td>Reporter</td>
<td>Status</td>
<td>Resolved In</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>---------</td>
<td>-------------</td>
<td>----------</td>
<td>--------</td>
<td>-------------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FE-3732</td>
<td>Switching from tag to trunk results in a 500 Server Error</td>
<td>Conor MacNeill [Atlassian]</td>
<td>Rene Verschoor [Atlassian]</td>
<td>Fixed</td>
<td>Apr 18, 2012</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CRUC-6125</td>
<td>after replying to a comment, the form to post another reply has &quot;discard&quot; and &quot;keep as draft&quot; disabled</td>
<td>Tom Davies [Atlassian]</td>
<td>Geoff Crain [Atlassian]</td>
<td>Fixed</td>
<td>May 08, 2012</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CRUC-6052</td>
<td>Crucible sometime</td>
<td>Tom Davies</td>
<td>Borek Bernard</td>
<td>Fixed</td>
<td>Apr 10, 2012</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### CRUC-5928
**Cannot create Anchor Patch for GIT**

<table>
<thead>
<tr>
<th>Assignee</th>
<th>Reporter</th>
<th>Priority</th>
<th>Status</th>
<th>Created</th>
<th>Updated</th>
</tr>
</thead>
</table>

### CRUC-5779
**Abandoned reviews cannot be searched for after a Crucible reindex**

<table>
<thead>
<tr>
<th>Assignee</th>
<th>Reporter</th>
<th>Priority</th>
<th>Status</th>
<th>Created</th>
<th>Updated</th>
</tr>
</thead>
</table>

### CRUC-4188
**Improve logging of database errors - java.sql.BatchUpdateException: Batch entry 0... was aborted. Call getNextException to see the cause.**

<table>
<thead>
<tr>
<th>Assignee</th>
<th>Reporter</th>
<th>Priority</th>
<th>Status</th>
<th>Created</th>
<th>Updated</th>
</tr>
</thead>
</table>

### From 2.7.11 to 2.7.12

**16 April 2012**

This is a bug fix release. The complete list of issues is below.

### JIRA Issues (9 issues)

<table>
<thead>
<tr>
<th>Type</th>
<th>Key</th>
<th>Summary</th>
<th>Assignee</th>
<th>Reporter</th>
<th>Priority</th>
<th>Status</th>
<th>Resolved</th>
<th>Created</th>
<th>Updated</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="bug" /></td>
<td>FE-3974</td>
<td>Add the new system</td>
<td>Pierre-Etienne Poirot</td>
<td>Pierre-Etienne Poirot</td>
<td>-</td>
<td>Closed</td>
<td>Fixed</td>
<td>Feb 23, 2012</td>
<td>Apr 16, 2012</td>
</tr>
</tbody>
</table>
From 2.7.10 to 2.7.11

27 February 2012

This is a bug fix release. The complete list of issues is below.

### JIRA Issues (26 issues)

<table>
<thead>
<tr>
<th>Type</th>
<th>Key</th>
<th>Summary</th>
<th>Assignee</th>
<th>Reporter</th>
<th>Priority</th>
<th>Status</th>
<th>Resolution</th>
<th>Created</th>
<th>Updated</th>
</tr>
</thead>
<tbody>
<tr>
<td>🌱</td>
<td>FE-3941</td>
<td>Add a limit for the number of paths to index to a changeset</td>
<td>Matthew Watson</td>
<td>Matthew Watson</td>
<td></td>
<td>🏷️ Closed</td>
<td>Fixed</td>
<td>Feb 09, 2012</td>
<td>Feb 26, 2012</td>
</tr>
<tr>
<td>🌱</td>
<td>FE-3937</td>
<td>Add properties to stop branch and tag point ancestry detection for SVN</td>
<td>Matthew Watson</td>
<td>Matthew Watson</td>
<td>🏷️ Closed</td>
<td></td>
<td>Fixed</td>
<td>Feb 07, 2012</td>
<td>Feb 26, 2012</td>
</tr>
<tr>
<td>Issue</td>
<td>Summary</td>
<td>Assigned to</td>
<td>Resolution</td>
<td>Status</td>
<td>Created</td>
<td>Fixed</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>---------</td>
<td>-------------</td>
<td>------------</td>
<td>--------</td>
<td>---------</td>
<td>-------</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FE-3958</td>
<td>Content Indexing: Filter the paths to index before processing the list twice.</td>
<td>Matthew Watson</td>
<td>Close</td>
<td>Fixed</td>
<td>Feb 15, 2012</td>
<td>Feb 21, 2012</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FE-3957</td>
<td>Lucene re-index is only triggered for cross-repo until after first commit</td>
<td>Matthew Watson</td>
<td>Close</td>
<td>Fixed</td>
<td>Feb 15, 2012</td>
<td>Feb 16, 2012</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FE-3956</td>
<td>Content Indexing: Add an option to only index trunk, not root and trunk</td>
<td>Matthew Watson</td>
<td>Close</td>
<td>Fixed</td>
<td>Feb 15, 2012</td>
<td>Feb 26, 2012</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FE-3955</td>
<td>ReIndex: Only delete all the content docs if there are &gt;0 docs in the index</td>
<td>Matthew Watson</td>
<td>Close</td>
<td>Fixed</td>
<td>Feb 15, 2012</td>
<td>Feb 19, 2012</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FE-3953</td>
<td>Per user LoC figures are meaningless when</td>
<td>Tom Davies [Atlassian]</td>
<td>Close</td>
<td>Fixed</td>
<td>Feb 14, 2012</td>
<td>Feb 15, 2012</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ID</td>
<td>Summary</td>
<td>Reporter</td>
<td>Assignee</td>
<td>Status</td>
<td>Created</td>
<td>Fixed</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>--------------------------------------------------------------------------</td>
<td>---------------------------</td>
<td>---------------------------</td>
<td>----------</td>
<td>---------------</td>
<td>--------------</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FE-3952</td>
<td>Breadcru m links on the Users tab are incorrect when a branch is selected</td>
<td>Tom Davies [Atlassian]</td>
<td>Tom Davies [Atlassian]</td>
<td>Close d</td>
<td>Feb 14, 2012</td>
<td>Feb 14, 2012</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FE-3939</td>
<td>PHP syntax highlighting incorrectly matches keywords at the end of function names</td>
<td>Brendan Humphreys [Atlassian]</td>
<td>Chris Butler</td>
<td>Close d</td>
<td>Feb 08, 2012</td>
<td>Feb 08, 2012</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FE-3921</td>
<td>FishEye treats files as binary if they used to have a svn:mime-type property, even after the property has been removed</td>
<td>Michael Heemskerk [Atlassian]</td>
<td>Michael Heemskerk [Atlassian]</td>
<td>Close d</td>
<td>Jan 26, 2012</td>
<td>Feb 23, 2012</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Task ID</td>
<td>Description</td>
<td>Assigned To</td>
<td>Fixed By</td>
<td>Status</td>
<td>Date</td>
<td>Date</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---------</td>
<td>------------------------------------------------------------------------------</td>
<td>------------------------------</td>
<td>-------------------------------</td>
<td>---------</td>
<td>----------</td>
<td>----------</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FE-3905</td>
<td>Upgrade from 2.5-&gt;2.7 fires events for all commits and replays all smart commits</td>
<td>Matthew Watson</td>
<td>Matthew Watson</td>
<td>Fixed</td>
<td>Jan 20, 2012</td>
<td>Jan 23, 2012</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FE-3885</td>
<td>Smart Commits: Using the #time command with a complex time format (i.e. 2d 4h) only saves the 2d part</td>
<td>Nick Pellow [Atlassian]</td>
<td>Felipe Cuozzo [Atlassian]</td>
<td>Fixed</td>
<td>Jan 12, 2012</td>
<td>Mar 12, 2012</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FE-3855</td>
<td>Author column is misaligned in Annotatio View</td>
<td>Tom Davies [Atlassian]</td>
<td>Timothy Lusk</td>
<td>Fixed</td>
<td>Dec 09, 2011</td>
<td>Feb 19, 2012</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JIRA Key</td>
<td>Summary</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>----------</td>
<td>---------</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CRUC-6060</td>
<td>Work Day Plugin throwing an exception will cause the add comment transaction to be rolled back</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CRUC-6058</td>
<td>PagedSearch is not being injected with the request in some cases</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CRUC-6050</td>
<td>freemarker template for email notifications do not sufficiently null-check</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CRUC-6029</td>
<td>Reduce number of database calls when listing projects with Application</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>JIRA Key</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CRUC-6060</td>
<td>Michael Heemskerk [Atlassian]</td>
</tr>
<tr>
<td>CRUC-6058</td>
<td>Brendan Humphreys [Atlassian]</td>
</tr>
<tr>
<td>CRUC-6050</td>
<td>Brendan Humphreys [Atlassian]</td>
</tr>
<tr>
<td>CRUC-6029</td>
<td>Tom Davies [Atlassian] Nick Pellow [Atlassian]</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Status</th>
<th>Assigned To</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fixed</td>
<td>Brendan Humphreys [Atlassian]</td>
</tr>
<tr>
<td>Closed</td>
<td>Michael Heemskerk [Atlassian]</td>
</tr>
<tr>
<td>Fixed</td>
<td>Brendan Humphreys [Atlassian]</td>
</tr>
<tr>
<td>Closed</td>
<td>Tom Davies [Atlassian] Nick Pellow [Atlassian]</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Date</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feb 15, 2012</td>
<td>Feb 16, 2012</td>
</tr>
<tr>
<td>Feb 14, 2012</td>
<td>Mar 07, 2012</td>
</tr>
<tr>
<td>Feb 12, 2012</td>
<td>Feb 15, 2012</td>
</tr>
<tr>
<td>Jan 22, 2012</td>
<td>Feb 14, 2012</td>
</tr>
</tbody>
</table>
From 2.7.9 to 2.7.10

20 January 2012

This is a bug fix release. The complete list of issues is below.

<table>
<thead>
<tr>
<th>JIRA Issues</th>
<th>Type</th>
<th>Key</th>
<th>Summary</th>
<th>Assignee</th>
<th>Reporter</th>
<th>Priority</th>
<th>Status</th>
<th>Resolution</th>
<th>Created</th>
<th>Updated</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>🎯</td>
<td>FE-3903</td>
<td>Unable to see FishEye activity stream in JIRA Activity Stream gadget on dashboard when have Repository with no description</td>
<td>Unassigned</td>
<td>Matthew Watson</td>
<td>📡</td>
<td>🔧 Closed</td>
<td>Fixed</td>
<td>Jan 19, 2012</td>
<td>Jan 19, 2012</td>
</tr>
<tr>
<td></td>
<td>🎯</td>
<td>FE-3902</td>
<td>Issue Transition controls don't appear when closing a review</td>
<td>Unassigned</td>
<td>None</td>
<td>📡</td>
<td>🔧 Closed</td>
<td>Fixed</td>
<td>Jan 19, 2012</td>
<td>Jan 19, 2012</td>
</tr>
</tbody>
</table>
### From 2.7.8 to 2.7.9

19 January 2012

This is a bug fix release. The complete list of issues is below.

#### JIRA Issues (23 issues)

<table>
<thead>
<tr>
<th>Type</th>
<th>Key</th>
<th>Summar y</th>
<th>Assignee</th>
<th>Reporter</th>
<th>Priority</th>
<th>Status</th>
<th>Resolution</th>
<th>Created</th>
<th>Updated</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>FE-3896</td>
<td>FishEye wrongly removes admin privileges for a Crowd group when Crowd is not up when FishEye</td>
<td>Nick Pellow [Atlassian]</td>
<td>Alex Wei [Atlassian]</td>
<td></td>
<td>Close</td>
<td>Fixed</td>
<td>Jan 05, 2012</td>
<td>Jan 18, 2012</td>
</tr>
<tr>
<td>FE-3878</td>
<td>When processing perforce file revisions, only branch specs that have been added or modified since the last restart are considered</td>
<td>Anna Buttfield [Atlassian]</td>
<td>Anna Buttfield [Atlassian]</td>
<td>Fixed</td>
<td>Jan 08, 2012</td>
<td>Jan 09, 2012</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>#</td>
<td>Status</td>
<td>Summary</td>
<td>Assignee</td>
<td>Fixed</td>
<td>Created</td>
<td>Resolved</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>--------</td>
<td>-------------------------------------------------------------------------</td>
<td>---------------------</td>
<td>-----------------</td>
<td>----------</td>
<td>-----------</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FE-3817</td>
<td>Fixed</td>
<td>Clicking on repository in FisheyeAdministration generates error dialog in IE8</td>
<td>Anna Buttfield</td>
<td>Nov 04, 2011</td>
<td>Nov 04, 2011</td>
<td>Jan 17, 2012</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CRUC-6028</td>
<td>Fixed</td>
<td>Internal avatar on changes et comment is broken</td>
<td>Anna Buttfield</td>
<td>Jan 04, 2012</td>
<td>Jan 04, 2012</td>
<td>Jan 18, 2012</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CRUC-6013</td>
<td>If during review creation the content of a file revision can’t be uploaded (via FishEye), that file is regarded as having no content</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CRUC-6010</td>
<td>Crucible search doesn’t deal properly with author user names containing punctuation</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CRUC-6008</td>
<td>Stopped repositories cause NPEs in Crucible reviews</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CRUC-6003</td>
<td>Timelines only work with a small set of timezones</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CRUC-5996</td>
<td>Review creator is unavailable from within ReviewCreatorEvent handler</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CRUC-5995</td>
<td>In some places pre-2.0 review</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
From 2.7.7 to 2.7.8

30 November 2011

This is a bug fix release. The complete list of issues is below.

### JIRA Issues (12 issues)

<table>
<thead>
<tr>
<th>Type</th>
<th>Key</th>
<th>Summary</th>
<th>Assignee</th>
<th>Reporter</th>
<th>Priority</th>
<th>Status</th>
<th>Resolution</th>
<th>Created</th>
<th>Updated</th>
</tr>
</thead>
<tbody>
<tr>
<td>#</td>
<td>Number</td>
<td>Status</td>
<td>Created</td>
<td>Nov 27</td>
<td>Nov 29</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>----</td>
<td>---------</td>
<td>------------</td>
<td>---------</td>
<td>--------</td>
<td>--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FE-3846</td>
<td>crow admin groups get removed on startup</td>
<td>Fixed</td>
<td>Nov 27, 2011</td>
<td>Nov 29, 2011</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FE-3839</td>
<td>autocomplete dropdowns give inconsistent and sometimes wrong results (results missing)</td>
<td>Fixed</td>
<td>Nov 20, 2011</td>
<td>Nov 21, 2011</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FE-3775</td>
<td>Allowed Review Participants field doesn't work when some users are present</td>
<td>Fixed</td>
<td>Oct 11, 2011</td>
<td>Nov 29, 2011</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FE-3745</td>
<td>Changing Block Size triggers a reindex + restart, while only a restart is needed</td>
<td>Fixed</td>
<td>Sep 26, 2011</td>
<td>Nov 24, 2011</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FE-3040</td>
<td>FishEye Access Logging is not writing the username</td>
<td>Fixed</td>
<td>Apr 27, 2011</td>
<td>Nov 29, 2011</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FE-2473</td>
<td>Cannot</td>
<td>Fixed</td>
<td>Feb 18, 2011</td>
<td>Nov 29, 2011</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### From 2.7.6 to 2.7.7

**21 November 2011**

This is a bug fix release. The complete list of issues is below.

#### JIRA Issues (17 issues)

<table>
<thead>
<tr>
<th>Type</th>
<th>Key</th>
<th>Summery</th>
<th>Assignee</th>
<th>Reporter</th>
<th>Priority</th>
<th>Status</th>
<th>Resolution</th>
<th>Created</th>
<th>Updated</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="https://example.com" alt="bug" /></td>
<td>CRUC-1853</td>
<td>Display preferences menu doesn't disappear when filter menu clicked or menu item is selected</td>
<td>Jonathan Poh [Atlassian]</td>
<td>Seb Ruiz [Atlassian]</td>
<td></td>
<td>Close d</td>
<td>Fixed</td>
<td>Jul 23, 2009</td>
<td>Nov 16, 2011</td>
</tr>
<tr>
<td><img src="https://example.com" alt="bug" /></td>
<td>FE-3836</td>
<td>In SVN, if a tag is</td>
<td>Anna Buttfield</td>
<td>Anna Buttfield</td>
<td></td>
<td>Close</td>
<td>Fixed</td>
<td>Nov 16, 2011</td>
<td>Nov 17, 2011</td>
</tr>
</tbody>
</table>

Created by Atlassian in 2012. Licensed under a Creative Commons Attribution 2.5 Australia License.
<table>
<thead>
<tr>
<th>Issue ID</th>
<th>Summary</th>
<th>Creator</th>
<th>Assigned To</th>
<th>Resolution</th>
<th>Created</th>
<th>Updated</th>
</tr>
</thead>
</table>
### FE-3174
**When navigating forward in time in activity stream, items are shown earliest to latest down the page**
- **Assignee**: Michael Studman
- **State**: None
- **Resolution**: Close
- **Fixed**: Sep 22, 2010
- **Closed**: Jan 31, 2012

### FE-3155
**Browse/diff menu inactives look active and vice-versa**
- **Assignee**: Jonathan Poh
- **State**: None
- **Resolution**: Close
- **Fixed**: Dec 12, 2010
- **Closed**: Nov 16, 2011

### CRUC-5977
**Crucible 'active users' list loaded unnecessarily, making review creation and editing slow**
- **Assignee**: Tom Davies
- **State**: None
- **Resolution**: Close
- **Fixed**: Nov 14, 2011
- **Closed**: Nov 16, 2011

### CRUC-5972
**Side-by-Side Diffs in Crucible are misaligned. Line number columns are very long**
- **Assignee**: Adam Ahmed
- **State**: None
- **Resolution**: Close
- **Fixed**: Nov 10, 2011
- **Closed**: Nov 15, 2011

### CRUC-5957
**Code review reply comments saved as draft, are deleted when any**
- **Assignee**: Michael Studman
- **State**: None
- **Resolution**: Close
- **Fixed**: Nov 03, 2011
- **Closed**: Nov 11, 2011
### From 2.7.5 to 2.7.6

**8 November 2011**

This is a bug fix release. The complete list of issues is below.

#### JIRA Issues (19 issues)

<table>
<thead>
<tr>
<th>Type</th>
<th>Key</th>
<th>Summar</th>
<th>Assigne</th>
<th>Reporter</th>
<th>Priority</th>
<th>Status</th>
<th>Resolution</th>
<th>Created</th>
<th>Updated</th>
</tr>
</thead>
</table>

*Created by Atlassian in 2012. Licensed under a Creative Commons Attribution 2.5 Australia License.*
<table>
<thead>
<tr>
<th>ID</th>
<th>Summary</th>
<th>Assigned To</th>
<th>Reporter</th>
<th>Status</th>
<th>Fix Version</th>
<th>Created</th>
<th>Closed</th>
<th>Resolution</th>
</tr>
</thead>
<tbody>
<tr>
<td>FE-3816</td>
<td>Shut down the thread pool once the background upgrade tasks have been completed</td>
<td>Pierre-Etienne Poirot [Atlassian]</td>
<td>Pierre-Etienne Poirot [Atlassian]</td>
<td>Fixed</td>
<td>Nov 03, 2011</td>
<td>Nov 03, 2011</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FE-3611</td>
<td>Improve the way avatars are loaded to avoid unnecessary calls</td>
<td>Anna Buttfield [Atlassian]</td>
<td>Renan Battaglin [Atlassian]</td>
<td>Fixed</td>
<td>Jul 28, 2011</td>
<td>Nov 03, 2011</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FE-3823</td>
<td>If the total length of the branch names</td>
<td>Unassigned</td>
<td>Tom Davies [Atlassian]</td>
<td>Fixed</td>
<td>Nov 09, 2011</td>
<td>Nov 10, 2011</td>
<td></td>
<td></td>
</tr>
<tr>
<td>#</td>
<td>Issue</td>
<td>Description</td>
<td>Reporter 1</td>
<td>Reporter 2</td>
<td>Status</td>
<td>Assigned To</td>
<td>Created</td>
<td>Modified</td>
</tr>
<tr>
<td>-----</td>
<td>---------</td>
<td>-------------</td>
<td>------------</td>
<td>------------</td>
<td>----------</td>
<td>-------------</td>
<td>---------</td>
<td>----------</td>
</tr>
</tbody>
</table>
## From 2.7.4 to 2.7.5

### 21 October 2011

This is a bug fix release. The complete list of issues is below.

### JIRA Issues (19 issues)

<table>
<thead>
<tr>
<th>Type</th>
<th>Key</th>
<th>Summary</th>
<th>Assignee</th>
<th>Reporter</th>
<th>Priority</th>
<th>Status</th>
<th>Resolution</th>
<th>Created</th>
<th>Updated</th>
</tr>
</thead>
<tbody>
<tr>
<td>🎯</td>
<td>FE-3799</td>
<td>quotation marks being double escaped in latest comment text</td>
<td>Pierre-Etienne Poirot</td>
<td>Tim Pettersen</td>
<td></td>
<td></td>
<td></td>
<td>Nov 10, 2010</td>
<td>Nov 08, 2011</td>
</tr>
<tr>
<td>🎯</td>
<td>CRUC-5923</td>
<td>Unable to add files to review using &quot;Explore Repositories&quot; option</td>
<td>Geoff Crain</td>
<td>Piotr Stefan Stefaniak</td>
<td></td>
<td></td>
<td></td>
<td>Sep 26, 2011</td>
<td>Feb 02, 2012</td>
</tr>
<tr>
<td>🎯</td>
<td>CRUC-5953</td>
<td>Scala syntax highlighter highlights substrings of identifiers as keywords</td>
<td>Seb Ruiz [Atlassian]</td>
<td>Leif Wickland</td>
<td></td>
<td></td>
<td></td>
<td>Nov 01, 2011</td>
<td>Nov 01, 2011</td>
</tr>
</tbody>
</table>

### Correctly in some cases with JIRA user management

1. Some browsers drop params to server and Migration to Postgres fails
2. Unable to add files to review using "Explore Repositories" option
3. Some browsers drop params to server and Migration to Postgres fails
4. Some browsers drop params to server and Migration to Postgres fails

---

Documentation for Crucible 2.8

Created by Atlassian in 2012. Licensed under a Creative Commons Attribution 2.5 Australia License.
<table>
<thead>
<tr>
<th>ID</th>
<th>Title</th>
<th>Reporter</th>
<th>Fixed By</th>
<th>Status</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>FE-3788</td>
<td>FishEye logs lots of ApplicationPermissionException when synchronising Crowd users that have no permission on FishEye / Crucible</td>
<td>Michael Heemske rk [Atlassian]</td>
<td>d Close d</td>
<td>Fixed</td>
<td>Oct 19, 2011 Oct 19, 2011</td>
</tr>
<tr>
<td>#</td>
<td>Issue</td>
<td>Summary</td>
<td>Assignees</td>
<td>Resolution</td>
<td>Status</td>
</tr>
<tr>
<td>-----</td>
<td>-------</td>
<td>-------------------------------------------------------------------------</td>
<td>-----------------------------------------------</td>
<td>------------</td>
<td>---------</td>
</tr>
<tr>
<td>FE-3755</td>
<td>SVN: Paths under tag directories are displayed as deleted/empty directories (greyed out)</td>
<td>Michael Heemske rk [Atlassian]</td>
<td>[ ]</td>
<td>Close d</td>
<td>Fixed</td>
</tr>
<tr>
<td>FE-3752</td>
<td>Repository path, includes, excludes and hidden directories should be copied to newly created forks</td>
<td>Tom Davies [Atlassian]</td>
<td>[ ]</td>
<td>Close d</td>
<td>Fixed</td>
</tr>
<tr>
<td>FE-3751</td>
<td>'Store diffs' setting is not respected for initial imports of SVN repositories</td>
<td>Michael Heemske rk [Atlassian]</td>
<td>[ ]</td>
<td>Close d</td>
<td>Fixed</td>
</tr>
<tr>
<td>FE-3692</td>
<td>When pushing to a Git managed repo while another client is cloning, the client which is cloned freezes</td>
<td>Pierre-Etienne Poirot [Atlassian]</td>
<td>Pierre-Etienne Poirot [Atlassian]</td>
<td>Fixed</td>
<td>Sep 06, 2011</td>
</tr>
</tbody>
</table>
From 2.7.3 to 2.7.4

28 September 2011

This is a bug fix release. The complete list of issues is below.

<table>
<thead>
<tr>
<th>JIRA Issues (13 issues)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
</tr>
<tr>
<td>------</td>
</tr>
<tr>
<td><img src="image" alt=" " /></td>
</tr>
<tr>
<td><img src="image" alt=" " /></td>
</tr>
<tr>
<td>ID</td>
</tr>
<tr>
<td>------</td>
</tr>
<tr>
<td>FE-3742</td>
</tr>
<tr>
<td>FE-3738</td>
</tr>
<tr>
<td>CRUC-5 920</td>
</tr>
<tr>
<td>------------</td>
</tr>
<tr>
<td>Sep 25, 2011</td>
</tr>
<tr>
<td>Fixed</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CRUC-5 900</th>
<th>Showing only time for each change set (not from today) on Add Content to Review screen does not make sense</th>
</tr>
</thead>
<tbody>
<tr>
<td>Seb Ruiz [Atlassian]</td>
<td>Wojciech Seliga [Atlassian]</td>
</tr>
<tr>
<td>Sep 11, 2011</td>
<td>Oct 05, 2011</td>
</tr>
<tr>
<td>Fixed</td>
<td>Fixed</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CRUC-5 891</th>
<th>the crucible changes activity was never updated to the new layout, which means some css is broken</th>
</tr>
</thead>
<tbody>
<tr>
<td>Geoff Crain [Atlassian]</td>
<td>Geoff Crain [Atlassian]</td>
</tr>
<tr>
<td>Sep 06, 2011</td>
<td>Sep 20, 2011</td>
</tr>
<tr>
<td>Fixed</td>
<td>Fixed</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CRUC-5</th>
<th>Unable to</th>
</tr>
</thead>
<tbody>
<tr>
<td>Michael Conor</td>
<td>Fixed</td>
</tr>
</tbody>
</table>
From 2.7.2 to 2.7.3

20 September 2011

This is a bug fix release. The complete list of issues is below.

### JIRA Issues (2 issues)

<table>
<thead>
<tr>
<th>Type</th>
<th>Key</th>
<th>Summary</th>
<th>Assignee</th>
<th>Reporter</th>
<th>Priority</th>
<th>Status</th>
<th>Resolution</th>
<th>Created</th>
<th>Updated</th>
</tr>
</thead>
</table>

From 2.7.1 to 2.7.2

19 September 2011

This is a bug fix release. The complete list of issues is below.

### JIRA Issues (12 issues)

<table>
<thead>
<tr>
<th>Type</th>
<th>Key</th>
<th>Summary</th>
<th>Assignee</th>
<th>Reporter</th>
<th>Priority</th>
<th>Status</th>
<th>Resolution</th>
<th>Created</th>
<th>Updated</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>FE-3718</td>
<td>Eclipse</td>
<td>Unassign</td>
<td>Tim</td>
<td>![^2]![^2]!</td>
<td>Close</td>
<td>Fixed</td>
<td>Sep 14, 2011</td>
<td>Sep 19,</td>
</tr>
<tr>
<td>---------</td>
<td>---------------------------------------------------------------</td>
<td>-------------------------</td>
<td>-------------------------</td>
<td>--------</td>
<td>---------</td>
<td>---------------</td>
<td>-------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FE-3696</td>
<td>Web Hooks should be deleted when deleting a repository</td>
<td>Tom Davies [Atlassian]</td>
<td>Seb Ruiz [Atlassian]</td>
<td>Closed</td>
<td>Fixed</td>
<td>Sep 06, 2011</td>
<td>Sep 13, 2011</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Documentation for Crucible 2.8

From 2.7.0 to 2.7.1

**9 September 2011**

This is a bug fix release. The complete list of issues is below.

#### JIRA Issues (5 issues)

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>CRUC-5886</td>
<td><strong>If the current user is a review's author, then switches to a reviewer, they show up in the manual Send Reminder dialog.</strong></td>
<td>Adam Ahmed [Atlassian]</td>
<td>Adam Ahmed [Atlassian]</td>
<td>Closed</td>
<td>Fixed</td>
<td>Sep 06, 2011</td>
<td>Sep 13, 2011</td>
</tr>
<tr>
<td>CRUC-298</td>
<td><strong>Should be able to mark files as text when the scm reports binary.</strong></td>
<td>Unassigned Eduardo M</td>
<td>Fixed</td>
<td>Apr 09, 2010</td>
<td>Oct 11, 2011</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Crucible 2.7 Upgrade Guide

Below are some important notes on upgrading to **Crucible 2.7**. For details of the new features and improvements in this release, please read the [Crucible 2.7 Release Notes](DocCrucibleReleaseNotes.html).

On this page:

<table>
<thead>
<tr>
<th>Type</th>
<th>Key</th>
<th>Summary</th>
<th>Assignee</th>
<th>Reporter</th>
<th>Priority</th>
<th>Status</th>
<th>Resolution</th>
<th>Created</th>
<th>Updated</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>CRUC-3298</td>
<td>Should be able to mark files as text when the scm reports binary</td>
<td>Unassigned</td>
<td>Eduardo M</td>
<td>⚠️</td>
<td>⚠️ Close</td>
<td>Fixed</td>
<td>Apr 09, 2010</td>
<td>Oct 11, 2011</td>
</tr>
</tbody>
</table>
Upgrade Notes

Crucible 2.7

- **Crucible Confluence LightSCM Plugin has Been Removed from Crucible**
  Previously, Crucible shipped with a plugin to enable reviews against content in Confluence. Since Confluence 4's new editor will no longer support wiki markup, this plugin has been deprecated. The Crucible plugin source as well as the source for the Confluence SCM plugin will remain open-source and are available via the Plugin Exchange at the following locations:

<table>
<thead>
<tr>
<th>Crucible Plugin</th>
<th>Confluence Crucible Plugin</th>
</tr>
</thead>
</table>

- **Crucible 2.7 adds ActiveObjects support**
  ActiveObjects allows plugins to store configuration data in the database. Please note, any already scheduled backups will not have plugin data included during backups. Please modify your scheduled backup configuration appropriately to include ActiveObjects backups.

Confluence has a number of free and commercially available review and workflow plugins. We recommend taking a look at these, to implement a smooth review process for Confluence pages.

Upgrade Procedure

**Before you begin**

- Test your upgrades in your test environment before rolling into production.
- Back up your entire Crucible instance (see Backing up and restoring Crucible data), i.e.
  - If you are backing up your Crucible instance via the Admin interface, tick all of the 'Include' checkboxes (e.g. plugins, templates, uploads, SQL database, etc).
  - If you are backing up your Crucible instance using the command-line interface, do not use any exclusion options.

If you are already running a version of Crucible, please follow the instructions on Upgrading to a new version of Crucible.

Checking for Known Issues and Troubleshooting the Crucible Upgrade

If something is not working correctly after you have completed the steps above to upgrade your Crucible installation, please check for known Crucible issues and try troubleshooting your upgrade as described below:

- **Check for known issues.** Sometimes Atlassian finds out about a problem with the latest version of Crucible after the software is released. In such cases we publish information about the known issues in the Crucible Knowledge Base. Please check the Crucible 2.7 Known Issues in the Crucible Knowledge Base and follow the instructions to apply any necessary patches if necessary.

- **Did you encounter a problem during the Crucible upgrade?** Please refer to the guide to troubleshooting upgrades in the Crucible Knowledge Base.

- If you encounter a problem during the upgrade and cannot solve it, please create a support ticket and one of our support engineers will help you.
Crucible 2.6 Release Notes

6 June 2011

With great pleasure, Atlassian presents Crucible 2.6 with SQL Server support, improved Quick Search, much better handling of reviews for Subversion merge commits and HTML emails.

Highlights of this Release:

- New Quick Search
- HTML Emails for Reviews
- Dashboard and Navigation Improvements
- SQL Server Support
- Oracle Support
- Review Creation without Metadata Changes
- Improved Patch Anchoring
- And Even More Improvements

Responding to your Feedback:

🌟 Over 269 votes satisfied

Highlights of Crucible 2.6

1

New Quick Search

You'll be able to find information faster using Crucible's new Quick Search. The Quick Search helps you to find the exact information that you need with a click of a button. The search results are also easy to work with – the user interface has a clean look and feel and features wiki rendering for reviews, popup summaries for JIRA issues in commit messages and more.
HTML Emails for Reviews

A 20% time project has resulted in email notifications getting a dramatic facelift. Gone are the dreary plain text emails, replaced by much better-looking HTML emails. You’ll see the new emails in action for review notifications.
Dashboard and Navigation Improvements

The Dashboard and Header have been tweaked to simplify the user interface in this release. You won’t have to
click different tabs to find the activity stream, as it will always be displayed. We've moved the other functions to the header to remove the clutter and provide you with a more streamlined view.

SQL Server Support

We are happy to announce that Crucible 2.6 now supports Microsoft SQL Server 2005 and 2008.

Oracle Support

We previously announced beta support for the Oracle DBMS in Crucible 2.5. We are happy to announce that this is no longer in beta, and that the Oracle DBMS is officially supported for Crucible 2.6.

Review Creation without Metadata Changes

Creating a review from a Subversion commit will now exclude metadata-only changes to files. You won't get pesky `svn mergeinfo` properties cluttering your review screens anymore! We've also prevented changesets that are entirely svnprops changes (i.e. don't have any non-metadata changes) from being added to reviews.
Improved Patch Anchoring

We’ve revamped patch anchoring in this release. All the work has been done under the covers, to improve how a patch anchors to a repository. You should just notice that anchoring patches works better, when creating a review. And if something does go wrong, we’ve built in better error handling so that you’ll know what went wrong.

More...

And Even More Improvements

Visit our issue tracker to see the full list of improvements and bug fixes in FishEye and Crucible for this release.

Release Notices

- **Upgrading from a previous version of Crucible.** Upgrading Crucible should be fairly straightforward. We strongly recommend that you back up Crucible before upgrading. Please refer to the Crucible 2.6 Upgrade Guide for further essential information about your upgrade.
- **Known Issues.** Please check the important technical advisories on the front page of the Knowledge Base for information about any known issues for this release.

Crucible 2.6 Changelog

This page contains information about the Crucible 2.6 minor releases. FishEye license holders should also check the FishEye 2.6 Changelog.

See the Crucible 2.6 Release Notes for details of what’s new in 2.6.0.

> Please read the Crucible 2.6 Upgrade Guide before upgrading to any of the minor releases below.

On this page:

- From 2.6.6 to 2.6.7
- From 2.6.5 to 2.6.6
- From 2.6.4 to 2.6.5
- From 2.6.3 to 2.6.4
- From 2.6.2 to 2.6.3
- From 2.6.1 to 2.6.2
- From 2.6.0 to 2.6.1

From 2.6.6 to 2.6.7

31 January 2012
This is a bug fix release. The complete list of issues is below.

### JIRA Issues (1 issues)

<table>
<thead>
<tr>
<th>Type</th>
<th>Key</th>
<th>Summary</th>
<th>Assignee</th>
<th>Reporter</th>
<th>Priority</th>
<th>Status</th>
<th>Resolution</th>
<th>Created</th>
<th>Updated</th>
</tr>
</thead>
</table>

From 2.6.5 to 2.6.6

2 September 2011

This is a bug fix release. The complete list of issues is below.

### JIRA Issues (4 issues)

<table>
<thead>
<tr>
<th>Type</th>
<th>Key</th>
<th>Summary</th>
<th>Assignee</th>
<th>Reporter</th>
<th>Priority</th>
<th>Status</th>
<th>Resolution</th>
<th>Created</th>
<th>Updated</th>
</tr>
</thead>
<tbody>
<tr>
<td>🔄</td>
<td>CRUC-5853</td>
<td>Crucible ToDo gadget links don't work due to extra slashes in the url</td>
<td>Joe Xie [atlassian]</td>
<td>Jay Barra</td>
<td>🟥 Unresolved</td>
<td>Close</td>
<td>Fixed</td>
<td>Aug 01, 2011</td>
<td>Sep 08, 2011</td>
</tr>
</tbody>
</table>
From 2.6.4 to 2.6.5

24 August 2011

This is a FishEye bug fix release. For details, see the FishEye 2.6 Changelog.

### JIRA Issues (0 issues)

<table>
<thead>
<tr>
<th>Type</th>
<th>Key</th>
<th>Summary</th>
<th>Assignee</th>
<th>Reporter</th>
<th>Priority</th>
<th>Status</th>
<th>Resolution</th>
<th>Created</th>
<th>Updated</th>
</tr>
</thead>
</table>

From 2.6.3 to 2.6.4

22 August 2011

This is a FishEye bug fix release. For details, see the FishEye 2.6 Changelog.

### JIRA Issues (0 issues)

<table>
<thead>
<tr>
<th>Type</th>
<th>Key</th>
<th>Summary</th>
<th>Assignee</th>
<th>Reporter</th>
<th>Priority</th>
<th>Status</th>
<th>Resolution</th>
<th>Created</th>
<th>Updated</th>
</tr>
</thead>
</table>

From 2.6.2 to 2.6.3

9 August 2011

This is a bug fix release. The complete list of issues is below.

### JIRA Issues (5 issues)

<table>
<thead>
<tr>
<th>Type</th>
<th>Key</th>
<th>Summary</th>
<th>Assignee</th>
<th>Reporter</th>
<th>Priority</th>
<th>Status</th>
<th>Resolution</th>
<th>Created</th>
<th>Updated</th>
</tr>
</thead>
</table>
From 2.6.1 to 2.6.2

20 July 2011

This is a bug fix release. The complete list of issues is below.

### JIRA Issues (3 issues)

<table>
<thead>
<tr>
<th>Type</th>
<th>Key</th>
<th>Summary</th>
<th>Assignee</th>
<th>Reporter</th>
<th>Priority</th>
<th>Status</th>
<th>Resolution</th>
<th>Created</th>
<th>Updated</th>
</tr>
</thead>
</table>
### From 2.6.0 to 2.6.1

**22 June 2011**

This is a bug fix release. The complete list of issues is below.

<table>
<thead>
<tr>
<th>JIRA Issues</th>
<th>(8 issues)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Type</strong></td>
<td><strong>Key</strong></td>
</tr>
</tbody>
</table>
Crucible 2.6 Upgrade Guide

Below are some important notes on upgrading to Crucible 2.6. For details of the new features and improvements in this release, please read the Crucible 2.6 Release Notes.

On this page:

- Upgrade Notes
  - Crucible 2.6
  - Upgrade Procedure
  - Checking for Known Issues and Troubleshooting the Crucible Upgrade

Upgrade Notes

**Crucible 2.6**

- **Internet Explorer 7 and Java Platform 5 (JDK/JRE 1.5) no longer supported** — As per the End of Support Announcements for Crucible (published previously), we are no longer supporting Internet Explorer 7 and Java Platform 5 (JDK/JRE 1.5) in this release. See Supported Platforms page for the full list.
list of supported platforms for Crucible.

- **Disabling review creation without metadata** — Crucible 2.6 introduces review creation without metadata. This functionality is enabled by default. If you want to disable this functionality, you can do so by starting up Crucible with the following system property:
  -Dcrucible.detect.metadata.revision.changes=false

  For further details, read this FAQ: How do I force reviews to include SVN property changes?

---

**Upgrade Procedure**

⚠️ **Before you begin**
- Test your upgrades in your test environment before rolling into production.
- Back up your entire Crucible instance (see Backing up and restoring Crucible data), i.e.
  - If you are backing up your Crucible instance via the Admin interface, tick all of the 'Include' checkboxes (e.g. plugins, templates, uploads, SQL database, etc).
  - If you are backing up your Crucible instance using the command-line interface, do not use any exclusion options.

If you are already running a version of Crucible, please follow these instructions on Upgrading to a new version of Crucible.

**Checking for Known Issues and Troubleshooting the Crucible Upgrade**

If something is not working correctly after you have completed the steps above to upgrade your Crucible installation, please check for known Crucible issues and try troubleshooting your upgrade as described below:

- **Check for known issues**. Sometimes we find out about a problem with the latest version of Crucible after we have released the software. In such cases we publish information about the known issues in the Crucible Knowledge Base. Please check the Crucible 2.6 Known Issues in the Crucible Knowledge Base and follow the instructions to apply any necessary patches if necessary.

- **Did you encounter a problem during the Crucible upgrade?** Please refer to the guide to troubleshooting upgrades in the Crucible Knowledge Base.

- If you encounter a problem during the upgrade and cannot solve it, please create a support ticket and one of our support engineers will help you.

**RELATED TOPICS**

Crucible 2.6 Release Notes

**Crucible Release Summary**

**Crucible 2.7 — 7 September 2011**
- JIRA Transitions in Crucible
- Review Reminders
- Small Improvements
- More in release notes.

**Crucible 2.6 — 6 June 2011**
- New Quick Search
- HTML Emails for Reviews
- Dashboard and Navigation Improvements
- SQL Server Support
- Oracle Support

*Created by Atlassian in 2012. Licensed under a Creative Commons Attribution 2.5 Australia License.*
• Review Creation without Metadata Changes
• Improved Patch Anchoring
• More in release notes.

**Crucible 2.5 — 8 February 2011**
• Oracle (Beta)
• Redesigned Activity Stream
• Improved Header
• Comment Notification Batching
• More in release notes.

**Crucible 2.4 — 20 October 2010**
• Easier Application Linking
• Native Repository Access
• Starter Licenses
• Adding Changesets to Reviews Simplified
• User Interface Improvements
• Snippets Tweaks
• More in release notes.

**Crucible 2.3 — 26 May 2010**
• Snippet Reviews
• Changeset Discussions
• Mercurial SCM Alpha
• Review Coverage report
• Revamped Installation Process
• Gadgets
• More in release notes.

**Crucible 2.2 — 18 February 2010**
• Smart Pre-Commit (Patch) Support
• ‘No Moderator’ Reviews
• Wizard-Like Review Creation
• Integrated Timetracking Between Crucible and JIRA
• Edit Mode for Reviews
• More in release notes.

**Crucible 2.1 — 12 November 2009**
• Wiki Markup Rendering
• Progress Tracking
• Usability and Productivity Updates
• Streamlined JIRA Integration
• Review Time Tracking
• Review History Dialog
• "Blockers" Reports
• Threaded Comments
• Plugin Developer Tools
• More in release notes.

**Crucible 2.0 — 30 June 2009**
• Support for iterative reviews
• New User Interface
• Indicators for read/unread comments
• Enhanced JIRA integration
- More in release notes.

**Crucible 1.6 — 23 September 2008**
- Support for non-FishEye repositories
- Confluence page reviews
- Shared file system repositories
- Enhanced pre-commit reviews & image support
- Multiple admin users
- Expanded API
- More in release notes.

**Crucible 1.5 — 14 April 2008**
- Project Dashboard
- Filtered comments & defects search, with statistical summary
- Customisable email templates
- Improvements to Crucible Plugin API beta
- More in release notes.

**Crucible 1.2 — 5 Dec 2007**
- Reviews grouped into projects
- Customisable permission schemes
- Plugin API
- Enhancements to user management
- JIRA integration
- Crucible 1.2 includes FishEye 1.4
- More in release notes.

**Crucible 1.1 — 18 September 2007**
- Pre-commit review (patch review)
- Review participants can keep track of their progress through a review by marking each file as "done"
- Side-by-side diff mode within the Review display
- Syntax highlighting when displaying a diff
- More in release notes.

### Security advisories

#### Finding and reporting a security vulnerability

Atlassian's approach to releasing patches is detailed in [How to Report a Security Issue](#).

#### Publication of Bamboo security advisories

Atlassian's approach to publishing security advisories is detailed in [Security Advisory Publishing Policy](#).

#### Severity levels

Atlassian's scale for measuring security issues is detailed in [Severity Levels for Security Issues](#).

#### Our patch policy

Atlassian's approach to releasing patches is detailed in our [Security Patch Policy](#).

### Security advisories
FishEye and Crucible Security Advisory 2012-08-21

This advisory discloses security vulnerabilities that we have found in FishEye and/or Crucible and fixed in a recent version of FishEye and/or Crucible.

- **Customers who have downloaded and installed FishEye and/or Crucible** should upgrade their existing FishEye and/or Crucible installations to fix this vulnerability.
- **Atlassian OnDemand and JIRA Studio customers** are not affected by any of the issues described in this advisory.

Atlassian is committed to improving product security. The vulnerability listed in this advisory has been discovered by Atlassian, unless noted otherwise. The reporter may also have requested that we do not credit them.

If you have questions or concerns regarding this advisory, please raise a support request at [http://support.atlassian.com/](http://support.atlassian.com/).

In this advisory:

- **Elevation of privileges vulnerability**

Elevation of privileges vulnerability

Severity

Atlassian rates the severity level of this vulnerability as **Medium**, according to the scale published in [Severity Levels for Security Issues](http://support.atlassian.com/). The scale allows us to rank the severity as critical, high, moderate or low.

This is an independent assessment and you should evaluate its applicability to your own IT environment.

Description

We have identified and fixed a vulnerability in FishEye and Crucible that results from behaviour of certain third-party frameworks used in FishEye and Crucible. This vulnerability allows any attacker to:

- Set the FishEye and Crucible instance to allow anonymous access
- Set the FishEye and Crucible instance to allow anonymous signup

All versions of FishEye and Crucible up to and including 2.7.14 are affected by this vulnerability. The vulnerability is fixed in FishEye and/or Crucible 2.8.0 and later. This issue can be tracked at [FE-4222 - Authenticate](http://support.atlassian.com/) to see issue details and [CRUC-6188 - Authenticate](http://support.atlassian.com/) to see issue details.

The table below describes the FishEye and/or Crucible versions and the specific functionality affected by the vulnerabilities.

<table>
<thead>
<tr>
<th>FishEye and/or Crucible Vulnerability</th>
<th>Affected versions</th>
<th>Fixed Version</th>
<th>Issue Tracking</th>
</tr>
</thead>
</table>

Created by Atlassian in 2012. Licensed under a Creative Commons Attribution 2.5 Australia License.
Elevation of privileges

<table>
<thead>
<tr>
<th></th>
<th>2.5.x or earlier</th>
<th>2.5.9</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2.6.x</td>
<td>2.6.9</td>
</tr>
<tr>
<td></td>
<td>2.7.x</td>
<td>2.7.15, 2.8.0</td>
</tr>
</tbody>
</table>

**Note:** The email we sent out wrongly states that fixed versions are 2.5.8 and 2.6.7. FishEye and Crucible development team apologise for the mistake.

**Risk Mitigation**

If you cannot upgrade immediately, you can disable all access from the public Internet to your FishEye and/or Crucible instance to prevent external attacks.

**Fix**

**Upgrade**

The vulnerabilities and fix versions are described in the 'Vulnerability' section above.

We recommend that you upgrade to the latest version of FishEye and/or Crucible, if possible. For a full description of the latest version of FishEye and Crucible, see the FishEye release notes and Crucible release notes. You can download the latest version of FishEye and Crucible from the FishEye download centre and Crucible download centre.

There are no patches available.

**FishEye and Crucible Security Advisory 2012-05-17**

This advisory discloses a critical security vulnerability that exists in all versions of FishEye and Crucible up to and including 2.7.11.

- **Customers who have downloaded and installed** FishEye or Crucible should upgrade their existing FishEye and Crucible installations to fix this vulnerability.
- **Enterprise Hosted customers** need to request an upgrade by raising a support request at http://support.atlassian.com in the "Enterprise Hosting Support" project.
- **JIRA Studio and Atlassian OnDemand customers** are not affected by any of the issues described in this advisory.

Atlassian is committed to improving product security. The vulnerability listed in this advisory has been discovered by Atlassian, unless noted otherwise. The reporter may also have requested that we do not credit them.

If you have questions or concerns regarding this advisory, please raise a support request at http://support.atlassian.com/.

**In this advisory:**

- Critical XML Parsing Vulnerability
  - Severity
  - Description
  - Risk Mitigation
  - Fix
Critical XML Parsing Vulnerability

Severity

Atlassian rates the severity level of this vulnerability as critical, according to the scale published in Severity Levels for Security Issues. The scale allows us to rank the severity as critical, high, medium or low.

This is an independent assessment and you should evaluate its applicability to your own IT environment.

Description

We have identified and fixed a vulnerability in FishEye and Crucible that results from the way third-party XML parsers are used in FishEye and Crucible.

This vulnerability allows an attacker to:

- execute denial of service attacks against the FishEye and Crucible server, and
- read all local files readable to the system user under which FishEye and Crucible runs.

An attacker does not need to have an account with the affected FishEye or Crucible server to exploit this vulnerability.

All versions of FishEye and Crucible up to and including 2.7.11 are affected by this vulnerability. This issue can be tracked here: FE-4016 - Authenticate to see issue details

Risk Mitigation

We recommend that you upgrade your FishEye and Crucible installation to fix this vulnerability.

Alternatively, if you are not in a position to upgrade immediately, you should do all of the following until you can upgrade. Please note, these measures will only limit the impact of the vulnerability, they will not mitigate it completely.

- Disable access to the Remote, SOAP and XML-RPC APIs, if these remote APIs are not required. Note that remote API access is disabled by default. See enabling plugins for instructions.
- Disable public access (such as anonymous access and public signup) to your FishEye or Crucible instance until you have applied the necessary upgrade.
- Ensure that your FishEye/Crucible system user is restricted as described in best practices for configuring FishEye security.

Fix

Upgrade

Upgrade to FishEye and Crucible 2.7.12 or later which fixes this vulnerability. For a full description of these releases, see the FishEye and Crucible release notes. The following releases have also been made available to fix these issues in older FishEye and Crucible versions. You can download these versions from the FishEye and Crucible download centres.

- FishEye and Crucible 2.6.8 for FishEye and Crucible 2.6
- FishEye and Crucible 2.5.8 for FishEye and Crucible 2.5

Patches

There are no patches available for this vulnerability.

FishEye and Crucible Security Advisory 2012-01-31

This advisory discloses a CRITICAL security vulnerability that we have found in versions of FishEye and
Crucible from 2.0 up to and including 2.7.8. You need to upgrade your existing FishEye and Crucible installations to fix these vulnerabilities. Enterprise Hosted customers should request an upgrade by raising a support request at http://support.atlassian.com in the “Enterprise Hosting Support” project. Neither FishEye nor Crucible in Studio and Atlassian OnDemand are vulnerable to any of the issues described in this advisory.

Atlassian is committed to improving product security. The vulnerabilities listed in this advisory have been discovered by Atlassian, unless noted otherwise. The reporter may also have requested that we do not credit them.

If you have questions or concerns regarding this advisory, please raise a support request at http://support.atlassian.com.

In this advisory:

- Code Injection Vulnerability
  - Severity
  - Description
  - Vulnerability
  - Risk Mitigation
  - Fix

### Code Injection Vulnerability

#### Severity

Atlassian rates the severity level of this vulnerability as **CRITICAL**, according to the scale published in [Severity Levels for Security Issues](http://support.atlassian.com). The scale allows us to rank the severity as critical, high, medium or low.

#### Description

We have identified and fixed a code injection vulnerability in FishEye and Crucible caused by an underlying vulnerability in the third-party Webwork 2 framework. This vulnerability allows an attacker to run arbitrary Java code on a FishEye/Crucible server with user privileges of the FishEye/Crucible process. This vulnerability is a variant of a recently disclosed [Struts2 vulnerability](http://struts.apache.org/). The vulnerability exists in FishEye and Crucible pages accessible only by users with administrative privileges. It can be exploited with use of social engineering, e.g. by having the administrator click on a specially crafted link.

The maintainer of the original library can be contacted at [http://struts.apache.org/](http://struts.apache.org/).

#### Vulnerability

The table below describes the FishEye and Crucible versions and the specific functionality affected by the command injection vulnerability.

<table>
<thead>
<tr>
<th>FishEye and Crucible Component</th>
<th>Affected FishEye and Crucible Versions</th>
<th>Fixed Version</th>
<th>Issue Tracking</th>
</tr>
</thead>
<tbody>
<tr>
<td>WebWork 2</td>
<td>2.0 - 2.7.8</td>
<td>2.6.7</td>
<td>FE-3891</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2.7.9</td>
<td></td>
</tr>
</tbody>
</table>

#### Risk Mitigation

We recommend that you upgrade your FishEye and Crucible installations to fix this vulnerability.

Alternatively, if you are not in a position to upgrade immediately and you judge it necessary, you can restrict access to your instance of FishEye/Crucible by using a firewall.
Fix

FishEye and Crucible 2.6.7, 2.7.9 and later versions fix this issue. View the issue linked above for information on fix versions. For a full description of the latest version of FishEye and Crucible, see the FishEye and Crucible release notes. You can download the latest versions from the FishEye and Crucible download centres. There are no patches available for these issues.

FishEye and Crucible Security Advisory 2011-11-22

This advisory discloses a number of security vulnerabilities that we have found and fixed in versions of FishEye/Crucible earlier than 2.5.7. You need to upgrade your existing FishEye and Crucible installations to fix these vulnerabilities. Enterprise Hosted customers should request an upgrade by raising a support request at http://support.atlassian.com in the “Enterprise Hosting Support” project. Neither JIRA Studio nor Atlassian OnDemand are vulnerable to any of the issues described in this advisory.

Atlassian is committed to improving product security. The vulnerabilities listed in this advisory have been discovered by Atlassian, unless noted otherwise. The reporter may also have requested that we do not credit them.

If you have questions or concerns regarding this advisory, please raise a support request at http://support.atlassian.com.

In this advisory:

- **XSS Vulnerabilities**
  - **Severity**
  - **Risk Assessment**
  - **Vulnerabilities**
  - **Risk Mitigation**
  - **Fix**

- **Permission Verification Vulnerabilities**
  - **Severity**
  - **Risk Assessment**
  - **Vulnerability**
  - **Risk Mitigation**
  - **Fix**

**XSS Vulnerabilities**

**Severity**

Atlassian rates the severity level of these vulnerabilities as high, according to the scale published in Severity Levels for Security Issues. The scale allows us to rank the severity as critical, high, medium or low. These vulnerabilities are not critical.

**Risk Assessment**

We have identified and fixed a number of stored cross-site scripting (XSS) vulnerabilities which affect FishEye/Crucible instances, including publicly available instances (that is, internet-facing servers). XSS vulnerabilities allow an attacker to embed their own JavaScript into a FishEye/Crucible page. You can read more about XSS attacks on Wikipedia, at The Web Application Security Consortium and other places on the web.

**Vulnerabilities**

The table below describes the FishEye/Crucible versions and the specific functionality affected by the XSS vulnerabilities.
Risk Mitigation

We recommend that you upgrade your FishEye/Crucible installation to fix these vulnerabilities.

Alternatively, if you are not in a position to upgrade immediately and you judge it necessary, you can disable public signup to your instance as a temporary mitigation until you have applied the upgrade. For tighter access control, you can restrict access to trusted groups.

Fix

FishEye and Crucible 2.5.5 and later versions fix two these issues. View the issues linked above for information on fix versions. You can download the latest versions of FishEye and Crucible from the download centres (FishEye download centre, Crucible download centre).

There are no patches available to fix these vulnerabilities. You must upgrade your FishEye/Crucible installation.

Permission Verification Vulnerabilities

Severity

Atlassian rates the severity level of these vulnerabilities as medium, according to the scale published in Severity Levels for Security Issues. The scale allows us to rank the severity as critical, high, medium or low. These vulnerabilities are not critical.

Risk Assessment

We have identified and fixed two permission verification bugs which affect FishEye/Crucible instances, including publicly available instances (that is, internet-facing servers). These vulnerabilities allow users to view metadata for changesets and reviews, from repositories/projects that they do not have access to, via tooltips.

Vulnerability

The table below describes the FishEye/Crucible versions and the specific functionality affected by the vulnerabilities.

<table>
<thead>
<tr>
<th>FishEye/Crucible Feature</th>
<th>Affected FishEye/Crucible Versions</th>
<th>Fixed Version</th>
<th>Issue Tracking</th>
</tr>
</thead>
<tbody>
<tr>
<td>FishEye changeset tooltips</td>
<td>FishEye 2.4.6 to 2.5.6</td>
<td>FishEye 2.5.7</td>
<td>FE-3811</td>
</tr>
<tr>
<td>Crucible review tooltips</td>
<td>Crucible 2.4.6 to 2.5.6</td>
<td>Crucible 2.5.7</td>
<td>CRUC-5811</td>
</tr>
</tbody>
</table>
**Risk Mitigation**

We recommend that you upgrade your FishEye/Crucible installation to fix these vulnerabilities.

Alternatively, if you are not in a position to upgrade immediately and you judge it necessary, you can disable anonymous access to your instance. Logged-in users will still be able to view metadata that they do not have permission to view, but anonymous users will be prevented from accessing this information.

**Fix**

FishEye and Crucible 2.5.7 and later versions fix these two issues. View the issues linked above for information on earlier fix versions for each issue. For a full description of this release, see the [FishEye 2.5 changelog](http://support.atlassian.com) and [Crucible 2.5 Changelog](http://support.atlassian.com). You can download the latest version of FishEye and Crucible from the download centres ([FishEye download centre](http://support.atlassian.com), [Crucible download centre](http://support.atlassian.com)).

There are no patches available to fix these vulnerabilities. You must upgrade your FishEye/Crucible installation.

**FishEye and Crucible Security Advisory 2011-05-16**

This advisory announces a number of security vulnerabilities that we have found and fixed in recent versions of FishEye/Crucible. You need to upgrade your existing FishEye/Crucible installations to fix these vulnerabilities. Enterprise Hosted customers should request an upgrade by filing a ticket at [JIRA](http://support.atlassian.com). JIRA Studio is not vulnerable to any of the issues described in this advisory.

Atlassian is committed to improving product security. The vulnerabilities listed in this advisory have been discovered by Atlassian, unless noted otherwise. The reporter may also have requested that we do not credit them.

If you have questions or concerns regarding this advisory, please raise a support request at [http://support.atlassian.com/](http://support.atlassian.com/).

**In this advisory:**

- **XSS Vulnerabilities**
  - Severity
  - Risk Assessment
  - Vulnerability
  - Risk Mitigation
  - Fix

**XSS Vulnerabilities**

**Severity**

Atlassian rates the severity level of these vulnerabilities as **high**, according to the scale published in [Severity Levels for Security Issues](http://support.atlassian.com). The scale allows us to rank the severity as critical, high, moderate or low.

**Risk Assessment**

We have identified and fixed a number of cross-site scripting (XSS) vulnerabilities which may affect FishEye/Crucible instances, including publicly available instances (that is, internet-facing servers). XSS vulnerabilities potentially allow an attacker to embed their own JavaScript into a FishEye/Crucible page. You can read more about XSS attacks at [cgisecurity.com](http://support.atlassian.com), [The Web Application Security Consortium](http://support.atlassian.com) and other places on the web.

**Vulnerability**

The table below describes the FishEye/Crucible versions and the specific functionality affected by the XSS...
vulnerabilities.

<table>
<thead>
<tr>
<th>FishEye/Crucible Feature</th>
<th>Affected FishEye/Crucible Versions</th>
<th>Issue Tracking</th>
</tr>
</thead>
<tbody>
<tr>
<td>Crucible snippets</td>
<td>Crucible 2.4.5 to 2.5.0</td>
<td>CRUC-5734</td>
</tr>
<tr>
<td>Crucible author mapping</td>
<td>Crucible 2.4.5 to 2.5.0</td>
<td>CRUC-5735</td>
</tr>
<tr>
<td>Crucible changeset comments in search results</td>
<td>Crucible 2.3.0 to 2.5.0</td>
<td>CRUC-5736</td>
</tr>
<tr>
<td>Crucible comments search</td>
<td>Crucible 2.2.6 to 2.5.0</td>
<td>CRUC-5737</td>
</tr>
<tr>
<td>FishEye/Crucible dashboard - review activity</td>
<td>FishEye/Crucible 2.2.8 to 2.5.2</td>
<td>FE-3031</td>
</tr>
<tr>
<td>FishEye/Crucible reviews list</td>
<td>FishEye/Crucible 2.2.8 to 2.5.2</td>
<td>FE-3032</td>
</tr>
</tbody>
</table>

**Risk Mitigation**

We recommend that you upgrade your FishEye/Crucible installation to fix these vulnerabilities.

Alternatively, if you are not in a position to upgrade immediately and you judge it necessary, you can disable public signup to your instance until you have applied the upgrade. For even tighter control, you could restrict access to trusted groups.

**Fix**

FishEye/Crucible 2.5.4 fixes all of these issues. View the issues linked above for information on earlier fix versions for each issue. For a full description of this release, see the FishEye 2.5 changelog and Crucible 2.5 Changelog. You can download the latest version of FishEye/Crucible from the download centre (FishEye download centre, Crucible download centre).

There are no patches available to fix these vulnerabilities. You must upgrade your FishEye/Crucible installation.

Our thanks to Marian Ventuneac of http://www.ventuneac.net, who reported FE-3031 and FE-3032. We fully support the reporting of vulnerabilities and we appreciate it when people work with us to identify and solve the problem.

**FishEye and Crucible Security Advisory 2011-01-12**

This advisory announces a number of security vulnerabilities that we have found and fixed in recent versions of FishEye/Crucible. You need to upgrade your existing FishEye/Crucible installations to fix these vulnerabilities. Enterprise Hosted customers should request an upgrade by filing a ticket at http://support.atlassian.com. JIRA Studio is not vulnerable to any of the issues described in this advisory.

Atlassian is committed to improving product security. The vulnerabilities listed in this advisory have been discovered by Atlassian, unless noted otherwise. The reporter may also have requested that we do not credit them.

If you have questions or concerns regarding this advisory, please raise a support request at http://support.atlassian.com.
In this advisory:

- **XSS Vulnerabilities**
  - **Severity**
  - **Risk Assessment**
  - **Vulnerability**
  - **Risk Mitigation**
  - **Fix**

- **Administration password logged in debug log**
  - **Severity**
  - **Risk Assessment**
  - **Vulnerability**
  - **Risk Mitigation**
  - **Fix**

- **Review comment search returns comments that a user has no permission to view**
  - **Severity**
  - **Risk Assessment**
  - **Vulnerability**
  - **Risk Mitigation**
  - **Fix**

- **Anonymous global access exposes entire user list**
  - **Severity**
  - **Risk Assessment**
  - **Vulnerability**
  - **Risk Mitigation**
  - **Fix**

### XSS Vulnerabilities

#### Severity

Atlassian rates the severity level of these vulnerabilities as **high**, according to the scale published in [Severity Levels for Security Issues](#). The scale allows us to rank the severity as critical, high, moderate or low.

#### Risk Assessment

We have identified and fixed a number of cross-site scripting (XSS) vulnerabilities which may affect FishEye/Crucible instances, including publicly available instances (that is, internet-facing servers). XSS vulnerabilities potentially allow an attacker to embed their own JavaScript into a FishEye/Crucible page. You can read more about XSS attacks at [cogsecurity.com](http://cogsecurity.com), [The Web Application Security Consortium](http://www.wxappsecurity.org) and other places on the web.

#### Vulnerability

The table below describes the FishEye/Crucible versions and the specific functionality affected by the XSS vulnerabilities.

<table>
<thead>
<tr>
<th>FishEye/Crucible Feature</th>
<th>Affected FishEye/Crucible Versions</th>
<th>Issue Tracking</th>
</tr>
</thead>
<tbody>
<tr>
<td>Crucible review – linked JIRA</td>
<td>Crucible 2.0.1 – 2.3.7</td>
<td>CRUC-5307</td>
</tr>
<tr>
<td>Feature</td>
<td>Version Range</td>
<td>CRUC-XXXX</td>
</tr>
<tr>
<td>---------------------------------</td>
<td>-------------------</td>
<td>-----------</td>
</tr>
<tr>
<td>Crucible email reviews</td>
<td>Crucible 2.2.0 – 2.4.0</td>
<td>CRUC-5308</td>
</tr>
<tr>
<td>Crucible review reload</td>
<td>Crucible 2.2.0 – 2.4.2</td>
<td>CRUC-5309</td>
</tr>
<tr>
<td>Crucible edit review details screen</td>
<td>Crucible 2.2.0 – 2.3.7</td>
<td>CRUC-5345</td>
</tr>
<tr>
<td>FishEye repository configuration</td>
<td>FishEye 2.4.0</td>
<td>CRUC-5310</td>
</tr>
<tr>
<td>FishEye charts</td>
<td>FishEye 2.2.0 - 2.4.0</td>
<td>CRUC-5311</td>
</tr>
<tr>
<td>FishEye/Crucible code macro</td>
<td>FishEye/Crucible 2.2.0 – 2.4.0</td>
<td>CRUC-5312</td>
</tr>
<tr>
<td>FishEye/Crucible changeset page heading</td>
<td>FishEye/Crucible 2.3.2 – 2.4.0</td>
<td>CRUC-5313</td>
</tr>
</tbody>
</table>

**Risk Mitigation**

We recommend that you upgrade your FishEye/Crucible installation to fix these vulnerabilities.

Alternatively, if you are not in a position to upgrade immediately and you judge it necessary, you can disable public signup to your instance until you have applied the upgrade. For even tighter control, you could restrict access to trusted groups.

**Fix**

FishEye/Crucible 2.4.4 fixes all of these issues. View the issues linked above for information on earlier fix versions for each issue. For a full description of this release, see the FishEye 2.4 changelog and Crucible 2.4 Changelog. You can download the latest version of FishEye/Crucible from the download centre (FishEye download centre, Crucible download centre).

There are no patches available to fix these vulnerabilities. You must upgrade your FishEye/Crucible installation.

**Administration password logged in debug log**

**Severity**

Atlassian rates the severity level of these vulnerabilities as **high**, according to the scale published in Severity Levels for Security Issues. The scale allows us to rank the severity as critical, high, moderate or low.

**Risk Assessment**

We have identified and fixed a vulnerability in the FishEye/Crucible logging which may affect FishEye/Crucible instances, including publicly available instances (that is, internet-facing servers). This vulnerability allows administrator passwords to be logged in clear text when debug logging is enabled.

**Vulnerability**

This vulnerability affects FishEye and Crucible 2.2.0 to 2.4.0.

**Risk Mitigation**

We recommend that you upgrade your FishEye/Crucible installation to fix these vulnerabilities.

Alternatively, if you are not in a position to upgrade immediately and you judge it necessary, you can disable access logging. See Enabling Access Logging in FishEye and Enabling Access Logging in Crucible. You can...
also apply file restrictions to your log files. Note, this issue only occurs when DEBUG logging is turned on (off by default) when an administrator logs in.

Fix

FishEye/Crucible 2.4.2 and later fix this issue. For a full description of this release, see the FishEye 2.4 changelog and Crucible 2.4 Changelog. You can download the latest version of FishEye/Crucible from the download centre (FishEye download centre, Crucible download centre).

There are no patches available to fix this vulnerability. You must upgrade your FishEye/Crucible installation.

Review comment search returns comments that a user has no permission to view

Severity

Atlassian rates the severity level of these vulnerabilities as medium, according to the scale published in Severity Levels for Security Issues. The scale allows us to rank the severity as critical, high, moderate or low.

Risk Assessment

We have identified and fixed a vulnerability in the Crucible review comment search which may affect Crucible instances, including publicly available instances (that is, internet-facing servers). This vulnerability allows review comments to be displayed for projects that are not publicly viewable.

Vulnerability

This vulnerability affects Crucible 2.2.0 to 2.4.3.

Risk Mitigation

We recommend that you upgrade your FishEye/Crucible installation to fix these vulnerabilities.

Fix

FishEye/Crucible 2.2.5, 2.3.8 and 2.4.4 fix this issue. For a full description of this release, see the FishEye 2.4 changelog and Crucible 2.4 Changelog. You can download these versions of FishEye/Crucible via the download centre (FishEye download centre, Crucible download centre).

There are no patches available to fix this vulnerability. You must upgrade your FishEye/Crucible installation.

Anonymous global access exposes entire user list

Severity

Atlassian rates the severity level of these vulnerabilities as medium, according to the scale published in Severity Levels for Security Issues. The scale allows us to rank the severity as critical, high, moderate or low.

Risk Assessment

We have identified and fixed a vulnerability in the FishEye/Crucible anonymous global access which may affect FishEye/Crucible instances, including publicly available instances (that is, internet-facing servers). This vulnerability exposes the user list (usernames and emails) of a FishEye/Crucible instance for access when anonymous global access is enabled.

Vulnerability

This vulnerability affects FishEye and Crucible 2.2.0 to 2.4.3.
Risk Mitigation

We recommend that you upgrade your FishEye/Crucible installation to fix these vulnerabilities.

Alternatively, if you are not in a position to upgrade immediately and you judge it necessary, you can disable global anonymous access. See Configuring Anonymous Access.

Fix

FishEye/Crucible 2.2.6, 2.3.8 and 2.4.4 fix this issue. For a full description of this release, see the FishEye 2.4 changelog and Crucible 2.4 Changelog. You can download these versions of FishEye/Crucible via the download centre (FishEye download centre, Crucible download centre).

There are no patches available to fix this vulnerability. You must upgrade your FishEye/Crucible installation.

Crucible Security Advisory 2010-06-16

The 2.3.3 release of Crucible contains some security related fixes, which are part of the shared FishEye architecture. The following information for FishEye applies equally to Crucible.

The Crucible Download Centre has the updates for Crucible.

In this advisory:

- Remote Code Exploit Vulnerability
  - Severity
  - Risk Assessment
  - Vulnerability
  - Risk Mitigation
  - Fix

- Download Patches for Earlier FishEye / Crucible Versions
  - Patch for FishEye / Crucible 2.3.2
  - Patch for FishEye / Crucible 2.2.3

Remote Code Exploit Vulnerability

Severity

Atlassian rates this vulnerability as critical, according to the scale published in Severity Levels for Security Issues. The scale allows us to rank a vulnerability as critical, high, moderate or low.

Risk Assessment

We have identified and fixed a remote code exploit vulnerability which affects FishEye and Crucible instances.

Vulnerability

This vulnerability allows a motivated attacker to call remote code on the host server.

All versions of FishEye/Crucible up to version 2.3.2 are affected by this vulnerability.
This vulnerability has been discovered in XWork by OpenSymphony, a command pattern framework which is used by FishEye and Crucible.

**About the XWork Framework:**
- See the [OpenSymphony XWork page](http://www.opensymphony.com/xwork) for more information about XWork.

**Risk Mitigation**
We strongly recommend either upgrading or patching your FishEye/Crucible installation to fix this vulnerability. Please see the ‘Fix’ section below.

**Fix**
These issues have been fixed in FishEye 2.3.3 (see the changelog), which you can download from the download centre.

It has also been fixed in Crucible 2.3.3 (see the changelog), which you can download from the download centre.

Later versions will include protection from this vulnerability.

This fix is also provided as a patch for FishEye/Crucible 2.3.2 and 2.2.3, which you can download from links on this page. Customers on earlier point versions of FishEye/Crucible will have to upgrade to version 2.3.2 or 2.2.3 before applying the patch. Atlassian recommends you upgrade to FishEye/Crucible 2.3.3.

**Download Patches for Earlier FishEye / Crucible Versions**

These patch releases contain security fixes, which apply to the shared FishEye architecture that is the basis of both FishEye and Crucible.

Please note that these patches are for specific point versions of FishEye (2.3.2 and 2.2.3). If you are running an earlier version than these, you will need to upgrade to a version specifically addressed by one of these patches. Atlassian strongly recommends that you upgrade to FishEye 2.3.3 / Crucible 2.3.3 or later.

MD5 checksums are provided to allow verification of the downloaded files.

**Patch for FishEye / Crucible 2.3.2**

<table>
<thead>
<tr>
<th>File</th>
<th>FishEye / Crucible Version</th>
<th>Release Date</th>
<th>MD5 Checksum</th>
</tr>
</thead>
<tbody>
<tr>
<td>fisheye-crucible-2.3.2-patch1.zip</td>
<td>2.3.2</td>
<td>16th June, 2010</td>
<td>6fe98db821a6d26f26907688af2ccd84</td>
</tr>
</tbody>
</table>
Patch for FishEye / Crucible 2.2.3

<table>
<thead>
<tr>
<th>File</th>
<th>FishEye / Crucible Version</th>
<th>Release Date</th>
<th>MD5 Checksum</th>
</tr>
</thead>
<tbody>
<tr>
<td>fisheye-crucible-2.2.3-patch1.zip</td>
<td>2.2.3</td>
<td>16th June, 2010</td>
<td>6fe98db821a6d26f26907688af2ccd84</td>
</tr>
</tbody>
</table>

Our thanks to Meder Kydyraliev of the Google Security Team who discovered this vulnerability. Atlassian fully supports the reporting of vulnerabilities and appreciates it when people work with Atlassian to identify and solve the problem.

Crucible Security Advisory 2010-05-04

ℹ️ The 2.2.3 release of Crucible contains some security related fixes, which are part of the shared FishEye architecture. The following information for FishEye applies equally to Crucible.

The Crucible Download Centre has the updates for Crucible.

In this advisory:

- Admin Escalation Vulnerability
  - Severity
  - Risk Assessment
  - Vulnerability
  - Risk Mitigation
  - Fix

- XSS Vulnerabilities in FishEye
  - Severity
  - Risk Assessment
  - Vulnerability
  - Risk Mitigation
  - Fix

- Prevention of Brute Force Attacks
  - Severity
  - Risk Assessment
  - Vulnerability
  - Risk Mitigation
  - Fix

- Changed Behaviour in FishEye

- Download Patches for Earlier FishEye / Crucible Versions
  - Patch for FishEye / Crucible 2.1.4
  - Patch for FishEye / Crucible 2.0.6
  - Patch for FishEye 1.6.6
  - Patch for Crucible 1.6.6

Admin Escalation Vulnerability

Severity
Atlassian rates this vulnerability as **critical**, according to the scale published in *Severity Levels for Security Issues*. The scale allows us to rank a vulnerability as critical, high, moderate or low.

**Risk Assessment**

We have identified and fixed an admin escalation vulnerability, which affects FishEye instances. This vulnerability has security implications and is especially important for anyone running publicly accessible instances of FishEye.

**Vulnerability**

This vulnerability allows a motivated attacker to perform admin actions.

All versions of FishEye from version 1.6.0-beta2 (including 1.6.0) through to 2.2.1 are affected by these admin escalation vulnerabilities.

<table>
<thead>
<tr>
<th>Affected FishEye Versions</th>
<th>Fix Availability</th>
<th>More Details</th>
<th>Severity</th>
</tr>
</thead>
<tbody>
<tr>
<td>All versions up to and including 2.2.1</td>
<td>2.2.3 update, also available as patches for certain versions, listed on <a href="#">this page</a></td>
<td>This vulnerability allows a motivated attacker to perform admin actions.</td>
<td>Critical</td>
</tr>
</tbody>
</table>

**Risk Mitigation**

We strongly recommend either upgrading or patching your FishEye installation to fix this vulnerability. Please see the 'Fix' section below.

ℹ️ Note: If you are an Atlassian JIRA Studio customer, we have assessed that your system is secure and implemented additional protections for it.

**Fix**

These issues have been fixed in FishEye 2.2.3 (see the [changelog](#)), which you can download from the [download centre](#). Later versions will include protection from this vulnerability.

This fix is also provided as a patch for FishEye 2.1.4, 2.0.6 and 1.6.6, which you can download from [this page](#). Customers on earlier point versions of FishEye will have to upgrade to version 2.1.4, 2.0.6 or 1.6.6 before applying the patch. We recommend you upgrade to FishEye 2.2.3.

**XSS Vulnerabilities in FishEye**

**Severity**

Atlassian rates these vulnerabilities as **critical**, according to the scale published in *Severity Levels for Security Issues*. The scale allows us to rank a vulnerability as critical, high, moderate or low.

**Risk Assessment**
We have identified and fixed several cross-site scripting (XSS) vulnerabilities in FishEye, which may affect FishEye instances. These vulnerabilities have security implications and are especially important for anyone running publicly accessible instances of FishEye.

- The attacker might take advantage of the vulnerability to steal other users’ session cookies or other credentials, by sending the credentials back to the attacker’s own web server.
- The attacker’s text and script might be displayed to other people viewing a FishEye page. This is potentially damaging to your company’s reputation.

You can read more about XSS attacks at cgisecurity, CERT and other places on the web.

**Vulnerability**

All versions of FishEye are affected by these XSS vulnerabilities.

<table>
<thead>
<tr>
<th>Affected FishEye Versions</th>
<th>Fix Availability</th>
<th>More Details</th>
<th>Severity</th>
</tr>
</thead>
<tbody>
<tr>
<td>All versions up to and including 2.2.1</td>
<td>2.2.3 only</td>
<td>An attacker could take advantage of this vulnerability to steal other users’ session cookies or other credentials, or the attacker’s text and script might be displayed to other people viewing a FishEye page.</td>
<td>Critical</td>
</tr>
</tbody>
</table>

**Risk Mitigation**

We strongly recommend upgrading your FishEye installation to fix these vulnerabilities. Please see the ‘Fix’ section below.

**Fix**

These issues have been fixed in FishEye 2.2.3 (see the changelog), which you can download from the download centre.

**Prevention of Brute Force Attacks**

**Severity**

Atlassian rates this vulnerability as moderate, according to the scale published in Severity Levels for Security Issues.

**Risk Assessment**

We have improved the security of the following areas in FishEye:

- Prevention of brute force attacks by requiring users to solve a CAPTCHA test after a maximum number of repeated login attempts.
Vulnerability

We have identified and fixed a problem where FishEye allows an unlimited number of repeated login attempts, potentially opening FishEye to a brute force attack. Details of this improvement are summarised below.

<table>
<thead>
<tr>
<th>Affected FishEye Versions</th>
<th>Fix Availability</th>
<th>More Details</th>
<th>Severity</th>
</tr>
</thead>
<tbody>
<tr>
<td>All versions up to and including 2.2.1</td>
<td>2.2.3 only</td>
<td>FishEye allows an unlimited number of login attempts. This makes FishEye vulnerable to a brute force attack.</td>
<td>Moderate</td>
</tr>
</tbody>
</table>

Risk Mitigation

We recommend that you upgrade your FishEye installation to fix these vulnerabilities. Please see the 'fix' section below.

You can also prevent brute force attacks by following our guidelines on using Fail2Ban to limit login attempts.

Fix

This issue has been fixed in FishEye 2.2.3 (see the changelog). Later versions will include protection from this vulnerability. You can download FishEye 2.2.3 from the download centre.

Changed Behaviour in FishEye

In order to fix these issues, we have changed FishEye's behaviour as follows:

- After three consecutive failed login attempts, FishEye will display a CAPTCHA form asking the user to enter a given word when attempting to log in again. This will prevent brute force attacks via the login screen. The number of failed attempts needed to trigger the CAPTCHA testing is configurable. For more information, see the documentation for Brute force login protection.

⚠️ In addition, after three consecutive failed login attempts via the FishEye remote API, an error message will be returned. Human intervention will then be required to reset that login account, i.e. solve the CAPTCHA test via the login screen.

Download Patches for Earlier FishEye / Crucible Versions

⚠️ These patch releases contain security fixes, which apply to the shared FishEye architecture that is the basis of both FishEye and Crucible.

These patches fix the Admin Escalation vulnerability only. Please note that these patches are for specific older point versions of FishEye (2.1.4, 2.0.6 or 1.6.6). If you are running an earlier version than these, you will need to upgrade to a version specifically addressed by one of these patches. To update a more recent version of the product (2.1.5 through 2.2.1), please upgrade to FishEye 2.2.3 or later. Atlassian strongly recommends that you upgrade to FishEye 2.2.3 or later.

MD5 checksums are provided to allow verification of the downloaded files.

Created by Atlassian in 2012. Licensed under a Creative Commons Attribution 2.5 Australia License.
### Patch for FishEye / Crucible 2.1.4

<table>
<thead>
<tr>
<th>File</th>
<th>FishEye / Crucible Version</th>
<th>Release Date</th>
<th>MD5 Checksum</th>
</tr>
</thead>
<tbody>
<tr>
<td>fisheye-crucible-2.1.4-patch1.zip</td>
<td>2.1.4</td>
<td>4th May, 2010</td>
<td>6062fa2e1ad937295273 57fb97b0d2ea</td>
</tr>
</tbody>
</table>

### Patch for FishEye / Crucible 2.0.6

<table>
<thead>
<tr>
<th>File</th>
<th>FishEye / Crucible Version</th>
<th>Release Date</th>
<th>MD5 Checksum</th>
</tr>
</thead>
<tbody>
<tr>
<td>fisheye-crucible-2.0.6-patch1.zip</td>
<td>2.0.6</td>
<td>4th May, 2010</td>
<td>6aae75e2a5308121887b f9532473cf75</td>
</tr>
</tbody>
</table>

### Patch for FishEye 1.6.6

<table>
<thead>
<tr>
<th>File</th>
<th>FishEye Version</th>
<th>Release Date</th>
<th>MD5 Checksum</th>
</tr>
</thead>
<tbody>
<tr>
<td>fisheye-1.6.6-patch1.zip</td>
<td>1.6.6</td>
<td>4th May, 2010</td>
<td>210ef3358aff838617331b f22d331d7e</td>
</tr>
</tbody>
</table>

### Patch for Crucible 1.6.6

<table>
<thead>
<tr>
<th>File</th>
<th>Crucible Version</th>
<th>Release Date</th>
<th>MD5 Checksum</th>
</tr>
</thead>
<tbody>
<tr>
<td>crucible-1.6.6-patch1.zip</td>
<td>1.6.6</td>
<td>4th May, 2010</td>
<td>48e8e8ada0d2b3f8671 459051df1120</td>
</tr>
</tbody>
</table>

To acquire all of the fixes on this page, upgrade to FishEye 2.2.3, which you can download from the [download centre](http://download."

### Crucible FAQ
Crucible FAQ

Answers to frequently asked questions about configuring and using Crucible.

- **General FAQs**
  - Can I deploy Crucible or FishEye as a WAR?
  - Does Crucible support SSL (HTTPS)?
  - How do I force reviews to include SVN property changes?
  - How to Automate Daily Crucible Backups

- **Licensing FAQ**
  - What happens if I decide to stop using FishEye with Crucible
  - Do I need a FishEye licence to run Crucible?
  - Advantages of Native Repository Access over lightSCM plugins

- **Support Policies**
  - Bug Fixing Policy
  - How to Report a Security Issue
  - New Features Policy
  - Patch Policy
  - Security Advisory Publishing Policy
  - Security Patch Policy
  - Severity Levels for Security Issues

- **Troubleshooting**
  - Crucible freezes unexpectedly
  - JIRA Integration Issues
  - Problems with very long comments and MySQL migration
  - Increasing the session timeout

Most setup issues are likely to be related to the FishEye component of Crucible. Refer to the FishEye documentation:

- FishEye documentation
- FishEye FAQs
- Top Evaluator Questions
  - Can Crucible add support for new repositories?
  - Can I purchase Crucible on it's own?
  - Can I trial Crucible without FishEye?
  - How can I do reviews from the file system?
  - How does Crucible help enforce compliance and auditability?
  - How do I convince my team of the benefits of code review?
  - How do I do pre-commit reviews?
  - How do I raise defects in JIRA?
  - How do I review patch diffs?
  - What user permissions and review security is available?

Do you still have a question, or need help with Crucible? Please create a support request.

General FAQs

<table>
<thead>
<tr>
<th>Crucible General FAQs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Can I deploy Crucible or FishEye as a WAR? — Unfortunately FishEye and Crucible cannot be deployed as a WAR.</td>
</tr>
<tr>
<td>Does Crucible support SSL (HTTPS)?</td>
</tr>
<tr>
<td>How do I force reviews to include SVN property changes?</td>
</tr>
<tr>
<td>How to Automate Daily Crucible Backups — Configuring Crucible backups is easy.</td>
</tr>
</tbody>
</table>
Can I deploy Crucible or FishEye as a WAR?

Unfortunately FishEye and Crucible cannot be deployed as a WAR. FishEye has some special needs for performance reasons that are not easily supported on third-party containers. Whilst this is an often requested feature, there are no immediate plans to provide a WAR version of FishEye or FishEye+Crucible. However the upcoming separate edition of Crucible (i.e. without FishEye) may at some stage be available as a WAR.

Does Crucible support SSL (HTTPS)?

Crucible does not have any built-in support for running over SSL via the HTTPS protocol. However, it is possible to setup a proxy web server to forward requests to Crucible. Please see the page on Integrating with Other Web Servers.

How do I force reviews to include SVN property changes?

Subversion (SVN) allows users to store arbitrary name/value pairs associated with files and directories, called SVN properties. These properties can be used by users for storing metadata and are also used by Subversion (e.g. for storing where code is branched from)

These name/value pairs can only be changed as part of a changeset / commit in Subversion. As such, you will have changesets with purely changes to SVN properties, or changesets with a mix of textual changes and changes to SVN properties. Crucible 2.6 introduced review creation without metadata to prevent SVN properties from being included in reviews.

This functionality is enabled by default, and affects the following functionality:

- Creating a review — The "Create Review" link is disabled in the activity stream, dashboard and changeset page for any changeset where all the file revisions only differ by SVN properties.
- Adding content to a review — SVN property-only changesets cannot be added to the review from the 'Add Content' dialog. Changesets with a mix of SVN property changes and textual changes can be added to the review, however only the file revisions with textual changes are added. File revisions with both textual changes and SVN property changes are always added.
  (Note, explicitly adding a particular file revision to a review is still permitted, regardless of whether it is a textual change or metadata-only change.)

If you want to disable this functionality, you can do so by starting up Crucible with the following system property:
-Dcrucible.detect.metadata.revision.changes=false

How to Automate Daily Crucible Backups

Configuring Crucible backups is easy. To set daily Crucible backups, open the administration page, click the 'Backup' link under 'System' on the left navigation bar, and simply follow the instructions set out on the Backing up and restoring Crucible data page.

Licensing FAQ

Crucible Licensing FAQ

- What happens if I decide to stop using FishEye with Crucible — Crucible can be run as an application alone, without FishEye. However, if you decide to stop using FishEye with Crucible, you will lose certain functionality and will need to make configuration changes.
- Do I need a FishEye licence to run Crucible? — FishEye and Crucible are separate products. They can be run separately, and they can also be run together.
- Advantages of Native Repository Access over lightSCM plugins

What happens if I decide to stop using FishEye with Crucible

Crucible can be run as an application alone, without
FishEye. However, if you decide to stop using FishEye with Crucible, you will lose certain functionality and will need to make configuration changes.

On this page:

- How do I run Crucible without FishEye?
- How is Crucible without FishEye different from using Crucible with FishEye?
  - Conducting Reviews
  - Viewing Repositories/Files
  - Charts
- Can I still use lightSCM plugins with Crucible?

How do I run Crucible without FishEye?

- Have a valid Crucible license but not a FishEye license
  To run Crucible without FishEye, you need to have a valid Crucible license but not a FishEye license. Crucible will actually use a "light" mode of FishEye when you do not have valid FishEye license. Light FishEye is bundled with Crucible and does not need to be installed separately. For more information on Crucible with light FishEye, see How is Crucible without FishEye different from using Crucible with FishEye? below.

- Reconnect your repositories
  Any repositories that you have currently defined in FishEye will not be visible in Crucible after removing FishEye. You will need to reconnect these repositories, as described in the FishEye documentation. Note, all repositories supported in FishEye are supported in light FishEye.

  Legacy "lightSCM" plugins, like the Crucible Subversion SCM plugin, will still work. However, the functionality will be limited compared to using Crucible with light FishEye. See the Can I still use lightSCM plugins with Crucible? section below for more information.

How is Crucible without FishEye different from using Crucible with FishEye?

The following changes in functionality will occur if you use Crucible without FishEye (i.e. use Crucible with "light" FishEye).

**Conducting Reviews**

- When using Iterative Reviews in Crucible, you will not be prompted when a new version of a file is available.

**Viewing Repositories/Files**

- Files and changesets displayed in activity streams (e.g. the dashboard activity stream) will not render as links to the relevant files/changesets.
- You will not be able to see your content roots and repositories associated with projects.
- You will no longer be able to see repository lists and browse repositories using the 'Source' tab.

**Charts**

- You will not be able to view charts or code metrics.

Can I still use lightSCM plugins with Crucible?

Legacy "lightSCM" plugins, like the Crucible Subversion SCM plugin, will still work with Crucible. However, we recommend that you use the "light" FishEye implementation that is bundled with Crucible, as it supersedes the
lightSCM plugins.

For more information, please read this FAQ: Advantages of Native Repository Access over lightSCM plugins.

Do I need a FishEye licence to run Crucible?

FishEye and Crucible are separate products. They can be run separately, and they can also be run together.

We recommend that you run Crucible together with FishEye. If you choose to run Crucible alone without FishEye, you will have access to your repositories via the "light" FishEye implementation bundled with Crucible. However, a number of FishEye's advanced features will not be available to you, including pre-caching repository content (for improved performance), the ability to search and browse through repositories and FishEye's activity graphs.

For more information, please read the following FAQ: What happens if I decide to stop using FishEye with Crucible

Advantages of Native Repository Access over lightSCM plugins

Prior to Crucible 2.4, running Crucible without FishEye required the use of "lightSCM" plugins (like the Crucible Subversion SCM plugin). From version 2.4, Crucible provides native repository access which supersedes Crucible's bundled lightSCM plugins. Third-party lightSCM plugins are unaffected and will continue to work with Crucible. The bundled plugins will still be available, so your existing configurations will also continue to work unchanged.

If you are currently using any of the bundled lightSCM plugins, we recommend that you migrate to using native repository access for the following reasons:

- Atlassian's lightSCM plugins (not lightSCM itself) are being deprecated, i.e. we will not update any of the bundled lightSCM plugins after the 2.4 release.
- It is easier for us to support and maintain a single implementation of our SCM interfaces, rather than support the standard FishEye access and the lightSCM implementations.
- Native repository access provides full support for SCMs for which there are no current lightSCM plugin implementations, including CVS and Mercurial.
- Native repository access provides additional functionality that is not available in the lightSCM plugins including:
  - Viewable commits in the activity streams (e.g. the dashboard activity stream).
  - Repositories administration via the administration console.
  - Easier review creation due to the ability to search and browse the repository using the full power of FishEye. For example, browsing for a file to add to a new review (see Selecting the Files for the Review).
- Improved performance of native repository access over the lightSCM plugins. The lightSCM plugins retrieve data on demand from the underlying repository, rather than using caches and indexes like FishEye and native repository access. Hence, Crucible with native repository access, whilst requiring an initial indexing phase, will be faster than Crucible with lightSCM plugins during day-to-day operations.
- Native repository access allows for migration to a full FishEye license in future, if desired. Your repositories can simply be re-indexed for full FishEye functionality and existing reviews will then be available on the full repository.

To change over from lightSCM plugins to native repository access:

1. Disable your lightSCM plugins via the Crucible Administration Console (Plugins’ link under the ‘Systems Settings’ section in the left menu).
   📢 Do not disable the SCM plugins for connecting to a Confluence instance or a file system. Native repository access does not include functionality to connect these (nor does FishEye alone), hence you will still need to use plugins.
2. Add native repositories for any repositories that are currently connected via lightSCM plugins. See the Fis
3. If you are using Subversion or Perforce, we recommend that you set a "start" revision for the changeover, unless you need to review old code. This will eliminate the need for native repository support to index old repository activity, getting you up and running quickly.

**Support Policies**

Welcome to the support policies index page. Here, you'll find information about how Atlassian Support can help you and how to get in touch with our helpful support engineers. Please choose the relevant page below to find out more.

- [Bug Fixing Policy](#)
- [How to Report a Security Issue](#)
- [New Features Policy](#)
- [Patch Policy](#)
- [Security Advisory Publishing Policy](#)
- [Security Patch Policy](#)
- [Severity Levels for Security Issues](#)

To request support from Atlassian, please raise a support issue in our online support system. To do this, visit [support.atlassian.com](http://support.atlassian.com), log in (creating an account if need be) and create an issue under Crucible. Our friendly support engineers will get right back to you with an answer.

**Bug Fixing Policy**

**Summary**

- Atlassian Support will help with workarounds and bug reporting.
- Critical bugs will generally be fixed in the next maintenance release.
- Non critical bugs will be scheduled according to a variety of considerations.

**Raising a Bug Report**

Atlassian Support is eager and happy to help verify bugs — we take pride in it! Please open a support request in our [support system](#) providing as much information as possible about how to replicate the problem you are experiencing. We will replicate the bug to verify, then lodge the report for you. We'll also try to construct workarounds if they're possible.

Customers and plugin developers are also welcome to open bug reports on our issue tracking systems directly. Use [http://jira.atlassian.com](http://jira.atlassian.com) for the stand-alone products and [http://studio.atlassian.com](http://studio.atlassian.com) for JIRA Studio and Atlassian OnDemand.

When raising a new bug, you should rate the priority of a bug according to our [JIRA usage guidelines](#). Customers [should watch](#) a filed bug in order to receive e-mail notification when a "Fix Version" is scheduled for release.

**How Atlassian Approaches Bug Fixing**

Maintenance (bug fix) releases come out more frequently than major releases and attempt to target the most critical bugs affecting our customers. The notation for a maintenance release is the final number in the version (ie the 1 in 3.0.1).

If a bug is critical (production application down or major malfunction causing business revenue loss or high numbers of staff unable to perform their normal functions) then it will be fixed in the next maintenance release.
provided that:

- The fix is technically feasible (i.e. it doesn't require a major architectural change).
- It does not impact the quality or integrity of a product.

For non-critical bugs, the developer assigned to fixing bugs prioritises the non-critical bug according to these factors:

- How many of our supported configurations are affected by the problem.
- Whether there is an effective workaround or patch.
- How difficult the issue is to fix.
- Whether many bugs in one area can be fixed at one time.

The developers responsible for bug fixing also monitor comments on existing bugs and new bugs submitted in JIRA, so you can provide feedback in this way. We give high priority consideration to security issues.

When considering the priority of a non-critical bug we try to determine a 'value' score for a bug which takes into account the severity of the bug from the customer's perspective, how prevalent the bug is and whether roadmap features may render the bug obsolete. We combine this with a complexity score (i.e. how difficult the bug is). These two dimensions are used when developers self serve from the bug pile.

Further reading

See Atlassian Support Offerings for more support-related information.

How to Report a Security Issue

Finding and Reporting a Security Vulnerability

If you find a security bug in the product, please open an issue on http://jira.atlassian.com in the relevant project.

- Set the priority of the bug to ‘Blocker’.
- Provide as much information on reproducing the bug as possible.
- Set the security level of the bug to ‘Developer and Reporters only’.

All communication about the vulnerability should be performed through JIRA, so that Atlassian can keep track of the issue and get a patch out as soon as possible.

If you discover a security vulnerability, please attempt to create a test case that proves this vulnerability locally before opening either a bug or a support issue. When creating an issue, please include information on how the vulnerability can be reproduced; see our Bug Fixing Policy for general bug reporting guidelines. We will prioritise fixing the reported vulnerability if your report has information on how the vulnerability can be exploited.

Further reading

See Atlassian Support Offerings for more support-related information.

New Features Policy

Summary

- We encourage and display customer comments and votes openly in our issue tracking system, http://jira.atlassian.com.
- We do not publish roadmaps.
- Product Managers review our most popular voted issues on a regular basis.
- We schedule features based on a variety of factors.
- Our Atlassian Bug Fixing Policy is distinct from our Feature Request process.
- Atlassian provides consistent updates on the top 20 feature/improvement requests (in our issue tracker...
How to Track what Features are Being Implemented

When a new feature or improvement is scheduled, the 'fix-for' version will be indicated in the JIRA issue. This happens for the upcoming release only. We maintain roadmaps for more distant releases internally, but because these roadmaps are often pre-empted by changing customer demands, we do not publish them.

How Atlassian Chooses What to Implement

In every major release we aim to implement highly requested features, but it is not the only determining factor. Other factors include:

- Direct feedback from face to face meetings with customers, and through our support and sales channels.
- Availability of staff to implement features.
- Impact of the proposed changes on the application and its underlying architecture.
- How well defined the requested feature is (some issues gain in popularity rapidly, allowing little time to plan their implementation).
- Our long-term strategic vision for the product.

How to Contribute to Feature Development

Influencing Atlassian’s release cycle

We encourage our customers to vote on feature requests in JIRA. The current tally of votes is available online in our issue tracking system, http://jira.atlassian.com. Find out if your improvement request already exists. If it does, please vote for it. If you do not find it, create a new feature or improvement request online.

Extending Atlassian Products

Atlassian products have powerful and flexible extension APIs. If you would like to see a particular feature implemented, it may be possible to develop the feature as a plugin. Documentation regarding the plugin APIs is available. Advice on extending either product may be available on the user mailing-lists, or at Atlassian Answers.

If you require significant customisations, you may wish to get in touch with our partners. They specialise in extending Atlassian products and can do this work for you. If you are interested, please contact us.

Further reading

See Atlassian Support Offerings for more support-related information.

Patch Policy

Atlassian will only provide software patches in extremely unusual circumstances. If a problem has been fixed in a newer release of the product, Atlassian will request that you upgrade your instance to fix the issue. If it is deemed necessary to provide a patch, a patch will be provided for the current release and the last maintenance release of the last major version only.

Patches are issued under the following conditions:

- The bug is critical (production application down or major malfunction causing business revenue loss or high numbers of staff unable to perform their normal functions).
- A patch is technically feasible (i.e., it doesn't require a major architectural change)
  OR
- The issue is a security issue, and falls under our Security Patch Policy.

Atlassian does not provide patches for non-critical bugs.
Provided that a patch does not impact the quality or integrity of a product, Atlassian will ensure that patches supplied to customers are added to the next maintenance release. Customers should watch a filed bug in order to receive e-mail notification when a "Fix Version" is scheduled for release.

Patches are generally attached to the relevant http://jira.atlassian.com issue.

Further reading

See Atlassian Support Offerings for more support-related information.

Security Advisory Publishing Policy

Publication of Security Advisories

When a security vulnerability in an Atlassian product is discovered and resolved, Atlassian will inform customers through the following mechanisms:

- We will post a security advisory in the latest documentation of the affected product at the same time as releasing a fix for the vulnerability. This applies to all security advisories, including severity levels of critical, high, medium and low.
- We will send a copy of all security advisories to the Technical Alerts mailing list for the product concerned.
  Note: To manage your email subscriptions and ensure you are on this list, please go to my.atlassian.com and click 'Communications Centre' near the top right of the page.
- If the person who reported the vulnerability wants to publish an advisory through some other agency, such as CERT, we will assist in the production of that advisory and link to it from our own.

Early warning of critical security vulnerabilities:

- If the vulnerability is rated critical (see our criteria for setting severity levels) we may send an early warning to the Technical Alerts' mailing list approximately one week before releasing the fix. This early warning is in addition to the security advisory itself, described above.
- However, if the vulnerability is publicly known or being exploited, we will release the security advisory and patches as soon as possible, potentially without early warning.

Further reading

See Atlassian Support Offerings for more support-related information.

Security Patch Policy

Product Security Patch Policy

Atlassian makes it a priority to ensure that customers' systems cannot be compromised by exploiting vulnerabilities in Atlassian products.

Scope

This page describes when and how we release security patches and security upgrades for our products. It does not describe the whole of disclosure process that we follow. It also excludes JIRA Studio, since JIRA Studio will always be patched by Atlassian without additional notifications.

Critical vulnerabilities

When a Critical security vulnerability is discovered by Atlassian or reported by a third party, Atlassian will do all of the following:

- Issue a new, fixed release for the current version of the affected product as soon as possible, usually in a few days.
• Issue a binary patch for the current release.
• Issue a binary patch for the latest maintenance release of the previous version of the product.
• Patches for older versions or releases normally will not be issued.

Patches will be attached to the relevant JIRA issue. You can use these patches as a "stop-gap" measure until you upgrade your installation in order to fully fix the vulnerability.

Non-critical vulnerabilities

When a security issue of a High, Medium or Low severity is discovered, Atlassian will do all of the following:

• Include the fix into the next scheduled release, both for the current and previous maintenance versions.
• Where practical, provide new versions of plugins or other components of the product that can be upgraded independently.

You should upgrade your installation in order to fix the vulnerability.

Other information

Severity level of vulnerabilities is calculated based on Severity Levels for Security Issues.

Visit our general Atlassian Patch Policy as well.

Examples

Example 1: A critical severity vulnerability is found in a (hypothetical current release) JIRA 5.3.2. The last bugfix release in 5.2.x branch was 5.2.3. In this case, a patch will be created for 5.3.2 and 5.2.3. In addition, new bugfix releases, 5.3.3 and 5.2.4, which are free from this vulnerability, will be created in a few days.

Example 2: A high or medium severity vulnerability is found in the same release as in the previous example. The fix will be included into the currently scheduled releases 5.3.3 and 5.2.4. Release schedule will not be brought forward and no patches will be issued. If the vulnerability is in a plugin module, then a plugin upgrade package may still be supplied.

Further reading

See Atlassian Support Offerings for more support-related information.

Severity Levels for Security Issues

Severity Levels

Atlassian security advisories include a severity level. This severity level is based on our self-calculated CVSS score for each specific vulnerability. CVSS is an industry standard vulnerability metric. You can learn more about CVSS at FIRST.org web site.

CVSS scores are mapped into the following severity ratings:

• Critical
• High
• Medium
• Low

An approximate mapping guideline is as follows:

<table>
<thead>
<tr>
<th>CVSS score range</th>
<th>Severity in advisory</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 – 2.9</td>
<td>Low</td>
</tr>
</tbody>
</table>
Below is a summary of the factors which illustrate types of vulnerabilities usually resulting in a specific severity level. Please keep in mind that this rating does not take into account details of your installation.

**Severity Level: Critical**

Vulnerabilities that score in the critical range usually have most of the following characteristics:

- Exploitation of the vulnerability results in root-level compromise of servers or infrastructure devices.
- The information required in order to exploit the vulnerability, such as example code, is widely available to attackers.
- Exploitation is usually straightforward, in the sense that the attacker does not need any special authentication credentials or knowledge about individual victims, and does not need to persuade a target user, for example via social engineering, into performing any special functions.

For critical vulnerabilities, is advised that you patch or upgrade as soon as possible, unless you have other mitigating measures in place. For example, if your installation is not accessible from the Internet, this may be a mitigating factor.

**Severity Level: High**

Vulnerabilities that score in the high range usually have some of the following characteristics:

- The vulnerability is difficult to exploit.
- Exploitation does not result in elevated privileges.
- Exploitation does not result in a significant data loss.

**Severity Level: Medium**

Vulnerabilities that score in the medium range usually have some of the following characteristics:

- Denial of service vulnerabilities that are difficult to set up.
- Exploits that require an attacker to reside on the same local network as the victim.
- Vulnerabilities that affect only nonstandard configurations or obscure applications.
- Vulnerabilities that require the attacker to manipulate individual victims via social engineering tactics.
- Vulnerabilities where exploitation provides only very limited access.

**Severity Level: Low**

Vulnerabilities in the low range typically have very little impact on an organisation's business. Exploitation of such vulnerabilities usually requires local or physical system access.

**Further reading**

See [Atlassian Support Offerings](https://confluence.atlassian.com/doc/support-offerings/) for more support-related information.

**Troubleshooting**

<table>
<thead>
<tr>
<th>Crucible Troubleshooting</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Crucible freezes unexpectedly</td>
</tr>
<tr>
<td>- JIRA Integration Issues</td>
</tr>
<tr>
<td>- Problems with very long comments and MySQL migration — Affects Version</td>
</tr>
</tbody>
</table>
Crucible Troubleshooting

The most common cause of FishEye/Crucible issues is an incorrect symbolic setup (trunk/branch/tag) for Subversion repositories. If you are using Subversion and your initial index is taking forever, double-check that your symbolic setup matches your repository.

FishEye runs with the default Java heap of 64 megabytes. This is sometimes problematic for FishEye, especially for Subversion repositories during the initial scan. You can give FishEye's JVM more memory by setting the FISHEYE_OPTS environment variable.

Starting Crucible with the command line options --debug --debug-perf will print a lot of information to Crucible's logs. This can give you an insight into what is happening and possibly where you are stuck. Attach these logs along with your config.xml to an Atlassian support ticket, to speed up your support request.

Crucible freezes unexpectedly

Issue Symptoms

If your Crucible 2.0 or 2.0.1 instance freezes unexpectedly, this could be caused by a known issue with Crucible and MySQL database technology.

This issue manifests itself in some Crucible pages returning a server timeout error. To identify the issue, check the Crucible error log. For this issue, the following output will appear in the error log:

```
2009-07-15 15:34:45,555 ERROR [btpool0-519] fisheye.app
HibernateUtil-commitTransaction - Commit fail msg-0:Could not execute JDBC batch update
2009-07-15 15:34:45,556 ERROR [btpool0-519] fisheye.app
HibernateUtil-commitTransaction - Commit fail msg-1:Lock wait timeout exceeded; try restarting transaction
2009-07-15 15:34:45,557 ERROR [btpool0-519] fisheye.app
HibernateUtil-commitTransaction - Commit failed rolling back.
...
...
Caused by: java.sql.BatchUpdateException: Lock wait timeout exceeded; try restarting transaction
   at com.mchange.v2.c3p0.impl.NewProxyPreparedStatement.executeBatch(NewProxyPreparedStatement.java:1723)
   at org.hibernate.jdbc.BatchingBatcher.doExecuteBatch(BatchingBatcher.java:48)
   at org.hibernate.jdbc.AbstractBatcher.executeBatch(AbstractBatcher.java:246)
   ... 163 more
```

The Crucible error log can be found under FISHEYE_INST/var/log/fisheye-error.log.YYYY-MM-DD.

See the JIRA issue for more information.

Workaround
Until the issue is solved, the suggested course of action is to restart your Crucible instance. This will return Crucible to normal operation.

The Crucible development team is actively working on a solution and this be part of an upcoming point release of Crucible.

Requesting Support

If you require assistance in resolving the problem, please [raise a support request](https://confluence.atlassian.com/display/Crucible/Support+) under the Crucible project.

**JIRA Integration Issues**

*Users are mapped to their own accounts when using Trusted Applications.*

If you (or the general account used for JIRA access, if not using Trusted Applications) do not have the permissions to carry out the JIRA actions linked from Crucible, an error will occur. Depending on the error returned from JIRA, Crucible may not display the error correctly or display it at all, simply reporting that “An error has occurred”. To investigate what the error was, you can access the Crucible debug log, named `fisheye-debug.log.YYYY-MM-DD` under the `dist.inst/var/log` folder of your Crucible installation. In the debug log, look for the date and time when your error took place. Here, you will be able to follow the links and see what error the JIRA instance was producing by clicking through to JIRA.

If you are using JIRA 4.0 you will not be able to create subtasks in versions of Crucible prior to 2.0.5. If you are affected by this bug, please upgrade to at least 2.0.6 (2.0.5 is affected by another bug [CRUC-2471](https://confluence.atlassian.com/display/Crucible/CRUC-2471)).

**Problems with very long comments and MySQL migration**

**Affects Version**

This issue was introduced in Crucible 2.0 and fixed in Crucible 2.1.

**Issue Symptoms**

There is a known issue with Crucible 2.0.x and very long comments when migrating your database to MySQL. In some circumstances, this might result in truncation of very long comments, causing data loss.

Depending on your MySQL configuration, you may see an error message like this while migrating to MySQL, causing the migration to fail:

```
java.sql.BatchUpdateException: Data truncation: Data too long for column 'cru_message' at row 1
java.sql.BatchUpdateException: Data truncation: Data too long for column 'cru_message' at row 1
```

ℹ️ You may not see the message if you are running MySQL with default settings.

For more information, see the [JIRA issue](https://confluence.atlassian.com/display/Crucible/CRUC-2471).

**Workaround**

If your data contains very long comments or review descriptions (longer than 21,845 multibyte unicode characters), consider avoiding use of MySQL until you can upgrade the product. Alternatively, use PostgreSQL...
or the default (built-in) HSQLDB database.

This issue is now resolved. This issue was introduced in Crucible 2.0 and fixed in Crucible 2.1.

Requesting Support

If you require assistance in resolving the problem, please raise a support request under the Crucible project.

Increasing the session timeout

Crucible comes with remember me functionality, i.e. so long as user hasn't logged out the computer will remember the user. So technically a user should not be logged out, unless the user has disabled the saving of cookies in their browser settings.

If the user has disabled cookies, and does not want to enable the saving of cookies, then, you can add:

```
<session-config>
  <session-timeout>120</session-timeout>
</session-config>
```

in your WEB-INF/web.xml file (see Jetty Documentation), which will increase the session timeout value to two hours.

Crucible Resources

Resources for Evaluators

- Free Trial
- Feature Tour

Resources for Administrators

- Crucible Knowledge Base
- Crucible FAQ
- Tips of the Trade
- Guide to Installing an Atlassian Integrated Suite
- The big list of Atlassian gadgets

Downloadable Documentation

- Crucible documentation in PDF, HTML or XML formats

Plugins

- Crucible Developer Documentation
- Atlassian Plugin Exchange

Support

- Atlassian Support
- Support Policies

Forums

- Crucible Forum
- Crucible Developers Forum
Mailing Lists

- Visit http://my.atlassian.com to sign up for mailing lists relating to Atlassian products, such as technical alerts, product announcements and developer updates.

Feature Requests

- Issue Tracker and Feature Requests for 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

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.

See 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.
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>Action</td>
<td>Description</td>
<td>Role(s)</td>
</tr>
<tr>
<td>----------</td>
<td>-------------------------------------------------------------------------------------------------</td>
<td>--------------------</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'</td>
</tr>
<tr>
<td></td>
<td></td>
<td>'Moderator'</td>
</tr>
<tr>
<td>'Summarise'</td>
<td>Ability to summarise a review. (Normally this would be done after all reviewers have completed their review.)</td>
<td>'Moderator'</td>
</tr>
<tr>
<td>'Create'</td>
<td>Ability to create a review.</td>
<td>All logged-in users</td>
</tr>
<tr>
<td>'Recover'</td>
<td>Ability to resurrect an abandoned (i.e. cancelled) review.</td>
<td>'Creator'</td>
</tr>
<tr>
<td></td>
<td></td>
<td>'Moderator'</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.
state

A Crucible review moves through the following states in the following sequence:

<table>
<thead>
<tr>
<th>State</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Draft</td>
<td>See <a href="#">Creating a Review</a></td>
</tr>
<tr>
<td>Require Approval</td>
<td>Relevant only when the moderator is not the creator. See <a href="#">Issuing a Review</a>.</td>
</tr>
<tr>
<td>Under Review</td>
<td>See <a href="#">Issuing a Review</a> and <a href="#">Reviewing the Code</a>.</td>
</tr>
<tr>
<td>Summarize</td>
<td>See <a href="#">Summarising and Closing the Review</a>.</td>
</tr>
<tr>
<td>Closed</td>
<td>See <a href="#">Summarising and Closing the Review</a>.</td>
</tr>
</tbody>
</table>

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>Details</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 moderator 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.

Contributing to the Crucible Documentation

Would you like to share your Crucible hints, tips and techniques with us and with other Crucible users? We welcome your contributions.

On this page:

- Blogging your Technical Tips and Guides – Tips of the Trade
- Contributing Documentation in Other Languages
- Updating the Documentation Itself
  - Getting Permission to Update the Documentation
  - Our Style Guide
  - How we Manage Community Updates

Blogging your Technical Tips and Guides – Tips of the Trade

Have you written a blog post describing a specific configuration of Crucible or a neat trick that you have discovered? Let us know, and we will link to your blog from our documentation. More...

Contributing Documentation in Other Languages

Have you written a guide to Crucible in a language other than English, or translated one of our guides? Let us know, and we will link to your guide from our documentation. More...
Updating the Documentation Itself

Have you found a mistake in the documentation, or do you have a small addition that would be so easy to add yourself rather than asking us to do it? You can update the documentation page directly.

Getting Permission to Update the Documentation

Please submit the Atlassian Contributor License Agreement.

Our Style Guide

Please read our short guidelines for authors.

How we Manage Community Updates

Here is a quick guide to how we manage community contributions to our documentation and the copyright that applies to the documentation:

- **Monitoring by technical writers.** The Atlassian technical writers monitor the updates to the documentation spaces, using RSS feeds and watching the spaces. If someone makes an update that needs some attention from us, we will make the necessary changes.

- **Wiki permissions.** We use wiki permissions to determine who can edit the documentation spaces. We ask people to sign the Atlassian Contributor License Agreement (ACLA) and submit it to us. That allows us to verify that the applicant is a real person. Then we give them permission to update the documentation.

- **Copyright.** The Atlassian documentation is published under a Creative Commons CC BY license. Specifically, we use a Creative Commons Attribution 2.5 Australia License. This means that anyone can copy, distribute and adapt our documentation provided they acknowledge the source of the documentation. The CC BY license is shown in the footer of every page, so that anyone who contributes to our documentation knows that their contribution falls under the same copyright.

RELATED TOPICS

Tips of the Trade
Author Guidelines
Atlassian Contributor License Agreement

Tips of the Trade

Below are some links to external blog posts and articles containing technical tips and instructions on setting up and using Crucible. This page presents an opportunity for customers and community authors to share information and experiences.

The references here are specific to Crucible and are technical ‘how to’ guides written by bloggers who use Crucible. For more general information on code review solutions, best practices and business cases, please refer to the Atlassian website.

⚠️ Please be aware that these are external blogs and articles.

Most of the links point to external sites, and some of the information is relevant to a specific release of Crucible. Atlassian provides these links because the information is useful and relevant at the time it was written. Please check carefully whether the information is still relevant when you read it, and whether it is relevant to your version of Crucible. **Unless explicitly stated**, Atlassian does not offer support for third-party extensions or plugins. The information in the linked blog posts has not been tested or reviewed by Atlassian. We recommend that you test all solutions on a test server before trying them on your production site.
On this page:

- Running Atlassian Crucible (or Fisheye) on Linux Startup

Administration

Running Atlassian Crucible (or Fisheye) on Linux Startup

- By: Jarrod, on blog 'The stupid people made me do it!
- About: How to get Crucible to start automatically when the server boots, via an initialisation script that runs on RedHat-like Linux boxes
- Date and Crucible version: 9 March 2009; Crucible 1.6
- Related documentation: Running Crucible as a Windows service

✅ Have you written a technical tip for Crucible?

Add a comment to this page, linking to your blog post or article. We will include it if the content fits the requirements of this page.

✅ Feedback?

Your first port of call should be the author of the linked blog post. If you want to let us know how useful (or otherwise) a linked post is, please add a comment to this page.

Other Sources of Information

- Crucible documentation
- Atlassian website
- Atlassian forums
- Atlassian Blog
- Crucible plugins

Crucible Documentation in Other Languages

Below are some links to Crucible documentation written in other languages. In some cases, the documentation may be a translation of the English documentation. In other cases, the documentation is an alternative guide written from scratch in another language. This page presents an opportunity for customers and community authors to share documentation that they have written in other languages.

⚠️ Please be aware that these are external guides.

Most of the links point to external sites, and some of the information is relevant to a specific release of Crucible. Atlassian provides these links because the information is useful and relevant at the time it was written. Please check carefully whether the information is still relevant when you read it, and whether it is relevant to your version of Crucible. The information in the linked guides has not been tested or reviewed by Atlassian.

On this page:

- No guides yet
Adding Your Own Guide to this Page

Have you written a guide for Crucible in another language? Add a comment to this page, linking to your guide. We will include it if the content fits the requirements of this page.

Giving Feedback about One of the Guides

If you have feedback on one of the guides listed above, please give the feedback to the author of the linked guide.

If you want to let us know how useful (or otherwise) one of these guides is, please add a comment to this page.

Other Sources of Information

Crucible documentation
Atlassian website
Atlassian blog
Crucible plugins