Contents

Crucible Documentation Home .......................................................... 5
Getting Started ................................................................................. 5
Supported platforms ........................................................................ 6
  End of Support Announcements for Crucible .................................... 8
  End of Support Announcement for IBM ClearCase ........................ 10
Installing Crucible on Windows ....................................................... 11
Running Crucible as a Windows service .......................................... 13
Installing Crucible on Linux and Mac ............................................ 16
Configuring JIRA Integration in the Setup Wizard ......................... 20
Starting to use Crucible ................................................................. 27
Using Crucible ................................................................................. 34
  Using the Crucible screens ............................................................ 35
    Using the Review Dashboard ......................................................... 37
    Browsing all reviews ................................................................ 39
    Browsing source files ................................................................ 40
    Crucible Icons ........................................................................... 41
    Searching Crucible ..................................................................... 42
    Viewing people’s statistics in Crucible ....................................... 47
    Browsing Projects ..................................................................... 51
      Viewing Project Statistics .......................................................... 53
    Changing your User Profile ....................................................... 55
    Viewing Your Favourites ............................................................. 55
    Using favourites ......................................................................... 57
    Using Keyboard Shortcuts in Crucible ........................................ 60
The Crucible workflow ...................................................................... 61
  Defining your workflow ................................................................ 66
  Roles and Status Classifications ................................................... 68
Creating a review ............................................................................. 69
  Creating a review from FishEye ..................................................... 72
  Creating a review from JIRA ......................................................... 73
  Creating a review from a URL ....................................................... 74
  Creating a Snippet Review ............................................................ 74
  Creating reviews from the command line .................................... 75
Adding content to the review .......................................................... 77
  Iterative reviews .......................................................................... 82
  Creating patch files for pre-commit reviews ................................ 84
Choosing reviewers .......................................................................... 91
Performing the review ...................................................................... 94
  Starting a review ......................................................................... 95
  Adding comments ........................................................................ 97
  Changeset discussions .................................................................. 99
Flagging defects ............................................................................. 100
Creating JIRA issues from the review ............................................ 100
Viewing reports .............................................................................. 102
  Review Coverage report .............................................................. 104
Completing your review ................................................................. 107
Sending a review’s comments via Email ........................................ 108
Using the Review History Dialog .................................................. 111
Using RSS feeds in Crucible .......................................................... 112
Using Wiki Markup in Crucible ....................................................... 112
Tracking Crucible Review Metrics ................................................ 117
  Using Progress Tracking .............................................................. 117
  Using Time Tracking .................................................................. 118
JIRA integration in Crucible ............................................................ 119
Summarising and closing the review .............................................. 122
Managing your reviews ................................................................. 124

Created by Atlassian in 2013. Licensed under a Creative Commons Attribution 2.5 Australia License.
User documentation for Crucible 3.1

Using Review Reminders ........................................... 124
Moving a Review to Another Project ............................... 124
Deleting a review .................................................. 125
Transitioning JIRA issues ......................................... 126

Crucible releases .................................................. 127
Crucible upgrade guide ........................................... 130
Upgrading from FishEye to Crucible .............................. 136
Crucible 3.1 release notes ....................................... 138
Crucible 3.0 release notes ....................................... 141
Crucible 2.10 release notes ..................................... 145
Crucible 2.10 changelog .......................................... 148
Crucible 2.10 upgrade guide .................................... 152
Crucible 2.9 release notes ....................................... 153
Crucible 2.9 changelog .......................................... 157
Crucible 2.9 upgrade guide .................................... 161
Crucible 2.8 Release Notes ...................................... 161
Crucible 2.8 Changelog ........................................... 165
Crucible 2.8 Upgrade Guide ...................................... 169
Crucible 2.7 Release Notes ...................................... 170
Crucible 2.7 Changelog ........................................... 172
Crucible 2.7 Upgrade Guide ...................................... 214
Crucible 2.6 Release Notes ...................................... 215
Crucible 2.6 Changelog ........................................... 219
Crucible 2.6 Upgrade Guide ...................................... 224
Crucible 2.5 Release Notes ...................................... 225
Crucible 2.5 Changelog ........................................... 229
Crucible 2.5 Upgrade Guide ...................................... 242

Crucible FAQ ....................................................... 243
Crucible Resources ................................................ 244
General FAQs ...................................................... 245
Can I deploy Crucible or FishEye as a WAR? ................. 245
Does Crucible support SSL (HTTPS)? ......................... 245
How do I force reviews to include SVN property changes? 245
How to Automate Daily Crucible Backups .................... 246
Licensing FAQ ...................................................... 246
What happens if I decide to stop using FishEye with Crucible
Do I need a FishEye licence to run Crucible? ................. 247
Advantages of Native Repository Access over lightSCM plugins 247
Support Policies .................................................... 248
Bug Fixing Policy ................................................ 248
How to Report a Security Issue .................................. 249
New Features Policy ............................................. 250
Patch Policy ....................................................... 251
Security Advisory Publishing Policy ......................... 251
Security Patch Policy ........................................... 251
Severity Levels for Security Issues ............................. 252
Troubleshooting ..................................................... 253
Crucible freezes unexpectedly .................................. 254
JIRA Integration Issues ......................................... 255
Problems with very long comments and MySQL migration 255
Contributing to the Crucible Documentation ................. 255
Crucible Documentation in Other Languages ............... 256
Glossary ............................................................. 257
approve ......................................................... 258
authors in Crucible .............................................. 258
code review ..................................................... 258
comment ......................................................... 258
creator ............................................................ 258
defect ........................................................... 258
moderator ......................................................... 258
participant ...................................................... 258
permission scheme ................................................ 258

Created by Atlassian in 2013. Licensed under a Creative Commons Attribution 2.5 Australia License.
Getting Started

Atlassian Crucible is the on-premises code review solution for enterprise teams. It allows your development teams to catch major defects, improve code architecture, or discuss desired improvements, without the need for meetings.

This section describes how to install, set up and get started with Crucible.

Related pages:
- Using Crucible
- Administering Crucible
- Crucible FAQ
- Crucible upgrade guide
System requirements

- Crucible is a Java web application, that works with all modern browsers. See our Supported platforms page to find out about system requirements.

Download and install Crucible

- Windows
- Mac
- Linux

Work with projects

Crucible integrates with Subversion, Git, Mercurial, CVS and Perforce. Find out how to connect to repositories, set up projects and create reviews.

Integrate Crucible with other Atlassian applications

As a first step, see JIRA integration in Crucible.

Read more about using Crucible

You are looking at the Crucible documentation. Browse using the tree in the panel on the left, or use the search at the top right.

Supported platforms

This page shows the supported platforms for Crucible 3.1.x and its minor releases.

Key: ✔️ = Supported ❌ = Not Supported

<table>
<thead>
<tr>
<th>Java Version</th>
<th>✔️</th>
<th>❌</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle JRE / JDK</td>
<td>✔️ 1.6</td>
<td>✔️ 1.7</td>
</tr>
<tr>
<td></td>
<td>❌</td>
<td></td>
</tr>
<tr>
<td>OpenJDK</td>
<td>✔️ 1.7 (Linux only)</td>
<td>✔️</td>
</tr>
</tbody>
</table>

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

You can download an Oracle Java Runtime for Windows/Linux/Solaris. On Mac OS X, the JDK is bundled with the operating system.

For the OpenJDK, download and install instructions for Linux flavours are at http://openjdk.java.net/install/.

We highly recommend that you use the Oracle JVM or OpenJDK (for Linux only), or use the default Mac OS X JVM. 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.
<table>
<thead>
<tr>
<th>Operating Systems</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Microsoft Windows</td>
<td>✔</td>
</tr>
<tr>
<td>Linux</td>
<td>✔</td>
</tr>
<tr>
<td>Apple Mac OS X</td>
<td>✔</td>
</tr>
</tbody>
</table>

- Crucible is a pure Java application and should run on any platform provided the requirements for the JRE or JDK are satisfied.
- Although Crucible can be run in virtualised environments, Atlassian is not yet able to provide technical support for performance-related problems in a virtualised environment. If you do choose to run Crucible in a VM, please ensure that you choose a VM with good IO throughput.
- On OS X Java 7 performs significantly better than Java 6

<table>
<thead>
<tr>
<th>Databases</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>MySQL</td>
<td>✔ MySQL Enterprise Server 5.x ✔ MySQL Community Server 5.x</td>
</tr>
</tbody>
</table>

Crucible supports a number of external databases, as listed. See the Crucible Database documentation for further details.

- PostgreSQL ✔ 8.2, 8.3, 8.4
- Oracle ✔ 11g

- HSQLDB (3) ✔ Bundled; for evaluation use only

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.

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

<table>
<thead>
<tr>
<th>Version Control Systems</th>
<th></th>
</tr>
</thead>
</table>
Subversion (SVN) | Server: 1.1, 1.2, 1.3, 1.4, 1.5, 1.6, 1.7  
Client: SVNKit (bundled & the default)  
Native JavaHL 1.7  
Native JavaHL 1.6  

FishEye 3.1, and later, supports SVN 1.7.x with the native JavaHL 1.7 client.  
FishEye 3.1, and later, do not support SVN 1.7.x with the native JavaHL 1.6 client.  
See Native support for SVN 1.7 for discussion.  
Subversion 1.8 support is currently being assessed. FishEye does not yet work for file:// access to repositories created with Subversion 1.8.  

CVS (and CVSNT) | All versions  

Perforce (P4) | Client version 2007.3 or later  
Server version 2005.1 or later  
The Server must support the ztag tagged protocol.  

Git | 1.7.1.1 or later  

Mercurial (Hg) | 1.5.1 or later  
Mercurial 2.1 has a bug that makes it incompatible with Crucible. Please use Mercurial 2.1.1 or later.  

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>Announcement date</th>
<th>Crucible End of Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>JIRA versions earlier than 5.0</td>
<td>27 August 2013</td>
<td>As of Crucible 3.2</td>
</tr>
<tr>
<td>MySQL 5.0</td>
<td>4 October 2011</td>
<td>January 2012</td>
</tr>
<tr>
<td>PostgreSQL 8.0 and 8.1</td>
<td>4 October 2011</td>
<td>January 2012</td>
</tr>
<tr>
<td>IBM ClearCase (all versions)</td>
<td>4 April 2011</td>
<td>4 April 2012 (announcement)</td>
</tr>
</tbody>
</table>

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 new versions of databases, web browsers etc. are released, the cost of supporting multiple platforms grows exponentially, making it harder to provide the level of support our customers have come to expect from us. Therefore, we no longer support platform versions marked as end-of-life by the vendor, or very old versions that are no longer widely used.

On this page (most recent announcements first):

- Deprecated Crucible 3.2 support for older versions of JIRA (announced 27 August 2013)
- Deprecated database support for Crucible (announced 4 October 2011)
- Deprecated web browsers for Crucible (announced 21 March 2011)
- Deprecated Java platforms for Crucible (announced 21 March 2011)
- Deprecated SCM repository support for Crucible (announced 4 April 2011)
Deprecated Crucible 3.2 support for older versions of JIRA (announced 27 August 2013)

Atlassian announces the deprecation of support for Crucible communication with older versions of Atlassian JIRA. We will stop supporting older versions of JIRA as follows:

- From Crucible 3.2, support for Crucible to JIRA communication for versions of JIRA earlier than 5.0, will end. Please note that communication from JIRA to Crucible will continue to work as it currently does. Crucible 3.2 is expected to be released late in 2013.

If you have questions or concerns regarding this announcement, please email eol-announcement at atlassian dot com.

Deprecated database support for Crucible (announced 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 (announced 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 (announced 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 (announced 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. After the specified date, Atlassian will not support the functionality in any version of Crucible.

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 features.
- After 4 April 2012, Atlassian will not support IBM ClearCase functionality in any version of Crucible.

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 not be providing any ClearCase-related support for any Crucible version after 4 April 2012, and
s removed 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...
1. 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.
2. 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, that is, something like C:\Java\bin. Don't use a trailing backslash. We recommend that JAVA_HOME should point to the Java executable specified in your PATH.
4. Now, in a new command prompt, try running '%JAVA_HOME%\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 `<Crucible home directory>` in these instructions. If you use FishEye and Crucible together, they run as one instance — see [Crucible and FishEye](#).
- 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 `<Crucible home directory>` — these should be entirely separate locations. If you do put the instance directory in the `<Crucible 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 `<Crucible home directory>\config.xml` file to the root of your new Crucible instance directory.

⚠️ 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 as a Windows Service](#).
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.

7. Start Crucible!

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

```
bin\stop.bat
```

Created by Atlassian in 2013. Licensed under a Creative Commons Attribution 2.5 Australia License.
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.

### On this page:

- Installing the Java Service Wrapper
- Setting Crucible environment variables for Windows Services
- Troubleshooting
  - Extracting files from wrapper.zip
  - Warning when using 64-bit Java JDK
  - Wrapper configuration and \-server parameter

### Related pages:

- Installing Crucible on Windows

---

**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 `<Crucible home directory>` (that is, the directory into which Crucible was originally installed). Note, the resulting folder structure should be similar to `<Crucible home directory>\wrapper` or `<Crucible home directory>\wrapper\bin`, etc and NOT `<Crucible 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 `<Crucible home directory>\wrapper\conf\wrapper.conf` file, replacing this:

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

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

```plaintext
# 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:
# Tell the Wrapper to log the full generated Java command line.
wrapper.java.command.loglevel=INF

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

4. Set the FISHEYE_INST environment variable (and other Crucible-specific environment variables) in the `<Crucible 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 `<Crucible 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 `<Crucible 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 Crucible 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 `<Crucible home directory>\wrapper\conf\wrapper.conf` file.

If you run into any problems starting the wrapper, you'll find its logs in `<Crucible 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.

```bash
# 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:

```bash
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:

```java
# 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 1 | 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 later (1.7.0 or later for OpenJDK).

If you don’t see a supported version of Java, then get Java...

Download and install the Oracle Java Platform JDK, or OpenJDK.

Now try running `java -version` again to check the installation. The version of Java should be 1.6.0 or later (1.7.0 or later for OpenJDK).

3. Check that the system can find Java

In a terminal, run this:

```
echo $JAVA_HOME
```

You should see a path something like:

<table>
<thead>
<tr>
<th>System</th>
<th>Path</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSX</td>
<td>/System/Library/Frameworks/JavaVM.frame work/Versions/CurrentJDK/Home/</td>
</tr>
<tr>
<td>Linux</td>
<td>/usr/lib/jvm/default-java</td>
</tr>
</tbody>
</table>

If you don’t see a path to the Java location, then set JAVA_HOME...
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 `<Crucible home directory>` in these instructions. If you use FishEye and Crucible together, they run as one instance, and use the same home directory – see Crucible and FishEye.
- 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:
### 3. Now copy the newly extracted `<Crucible home directory>/config.xml` file to the root of your new Crucible instance directory.

⚠️ You should not locate your Crucible instance directory inside the `<Crucible home directory>` — they should be entirely separate locations. If you do put the instance directory in the `<Crucible 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 `<Crucible home directory>` and run this:

```bash
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 `<Crucible 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

#### Related pages:
- Linking Crucible to JIRA
- User management limitations and recommendations
- JIRA integration in Crucible

### 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>
<tbody>
<tr>
<td>The setup wizard displays one of the following error messages:</td>
<td>The setup wizard failed to complete registration of the peer-to-peer application link with JIRA. JIRA integration is only partially configured.</td>
<td>Remove the partial configuration if it exists, try the ‘Connect to JIRA’ step again, and then continue with the setup. Detailed instructions are below.</td>
</tr>
</tbody>
</table>
  * 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 this `<application>` server at `<URL>` to this JIRA server at `<URL>`.
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.

The setup wizard displays the following error message:

- Error reloading Crowd authentication

The setup wizard has completed the integration of your application with JIRA, but is unable to start synchronizing the JIRA users with your application.

Restart your application. You should then be able to continue with the setup wizard. If this solution does not work, please contact Atlassian Support.

The setup wizard displays the following error message:

- An error occurred: `java.lang.IllegalArgumentException`: Could not create the application in JIRA/Crowd (code: 500). Please refer to the logs for details.

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 `<application>` URL.

Remove the partial configuration if it exists, try the 'Connect to JIRA' step again, and then continue with the setup. Detailed instructions are below.

No users can log in after you have set up the application with JIRA integration.

Possible causes:

- There are no users in the group that you specified on the 'Connect to JIRA' screen.
- For FishEye: There are no groups specified in the 'groups to synchronize' section of your administration console.
- For Stash: You may not have granted any JIRA groups or users permissions to log in to Stash.

Go to JIRA and add some usernames to the group.

- For FishEye: Go to the FishEye administration screens and specify at least one group to synchronize. The default is `jira-users`.
- For Stash: Grant the Stash User permission to the relevant JIRA groups on the Stash Global permissions page.

If this solution does not work, please contact Atlassian Support.

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

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.
      - 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:
### 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://www.example.com:8080/jira/)
  - [http://jira.example.com](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.

---

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 area.
  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://www.example.com:8080/jira/)
       - [http://jira.example.com](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 Stash to JIRA for user management](#).

---

**Notes**

*Created by Atlassian in 2013. Licensed under a [Creative Commons Attribution 2.5 Australia License](https://creativecommons.org/licenses/by/2.5/).*
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 introduces the basics of using Crucible. By the end, you’ll know how to:

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

### Assumptions

We’re assuming that:

- You have installed and started the latest version of Crucible. See the details at [Installing Crucible on Windows](#) or [Installing Crucible on Linux and Mac](#).
- You are using a supported browser.
- You have admin permission in Crucible.

### Add a repository

First up we’re going to add a repository to Crucible.

Go to the Admin area by clicking on the ‘cog’ at the top right and choosing **Administration**:

![Repository Management](image)

Click on **Add Existing...** in the **Repositories** listing of the Administration area.

![Repository Add Existing](image)

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.

**Repositories**

- **Native Repository Access (1)**
- **Plugins (4)**

Crucible natively supports Subversion, CVS, Perforce, Git and Mercurial repositories.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Location</th>
<th>State</th>
<th>Last Update</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>jgplot</td>
<td>Mercurial</td>
<td>JS charts library</td>
<td><a href="https://bitbucket.org/tleonello/jgplot/">https://bitbucket.org/tleonello/jgplot/</a></td>
<td>Running</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Create a project in Crucible**

Crucible comes with a default project, with the key CR, but you will probably want to create your own projects to contain your reviews. This is achieved in a couple of steps.

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

**Edit Project**

**Identification**

Name: IQ Reviews

Key: QCR

**Content**

Default Repository: jqplot

- Store the contents of files in reviews

**Permissions Scheme**

Permission Scheme: default

**Moderator**

- Enable the Moderator role for this project

Default Moderator: 

- Start typing a user name then press enter to select.

**Default Reviewers**

- By default, allow anyone to join reviews after creation

Users: 

- Start typing a user name then press enter to select.

You'll see your new project in the Projects listing.
Create a review

Now that you have your own project you can 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 Changesets** 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 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 will 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.

- Using the Crucible screens
  - Using the Review Dashboard
  - Browsing all reviews
  - Browsing source files
  - Crucible Icons
  - Searching Crucible
• Viewing people's statistics in Crucible
• Browsing Projects
  • Viewing Project Statistics
• Changing your User Profile
• Viewing Your Favourites
• Using favourites
• Using Keyboard Shortcuts in Crucible
• The Crucible workflow
  • Defining your workflow
  • Roles and Status Classifications
• Creating a review
  • Creating a review from FishEye
  • Creating a review from JIRA
  • Creating a review from a URL
  • Creating a Snippet Review
  • Creating reviews from the command line
• Adding content to the review
  • Iterative reviews
  • Creating patch files for pre-commit reviews
• Choosing reviewers
• Performing the review
  • Starting a review
  • Adding comments
  • Changeset discussions
  • Flagging defects
  • Creating JIRA issues from the review
  • Viewing reports
    • Review Coverage report
  • Completing your review
  • Sending a review's comments via Email
  • Using the Review History Dialog
  • Using RSS feeds in Crucible
  • Using Wiki Markup in Crucible
  • Tracking Crucible Review Metrics
    • Using Progress Tracking
    • Using Time Tracking
• JIRA integration in Crucible
• Summarising and closing the review
• Managing your reviews
  • Using Review Reminders
  • Moving a Review to Another Project
  • Deleting a review
• Transitioning JIRA issues

Using the Crucible screens

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

On this page:

• Dashboard
• Header
• Recent activity
• Related pages

Dashboard

The dashboard is the first screen you see when you log into FishEye/Crucible. The dashboard displays reviews and system activity related to you, and provides filtering for your recent repositories and projects. The dashboard can be accessed from anywhere in the application by clicking the FishEye/Crucible icon in the header.

Click View review dashboard to see more information about your reviews.

Screenshot: The Crucible dashboard (with FishEye), showing current reviews and recent activity
The table below explains the tabs in the Crucible header:

<table>
<thead>
<tr>
<th>Tab</th>
<th>Function</th>
<th>Appears</th>
</tr>
</thead>
<tbody>
<tr>
<td>Repositories</td>
<td>Displays contents of connected source repositories. The dropdown menu has links to recently visited repositories.</td>
<td>Only when FishEye is used with Crucible. Only when FishEye is used with Crucible for logged-in users.</td>
</tr>
<tr>
<td>Projects</td>
<td>Displays reviews and content from specific projects. The dropdown menu has links to recently visited projects.</td>
<td>All screens. All screens for logged-in users.</td>
</tr>
<tr>
<td>People</td>
<td>Displays metrics on the users of the Crucible instance. The dropdown menu has links to recently visited user pages.</td>
<td>All screens. All screens for logged-in users.</td>
</tr>
</tbody>
</table>
| Reviews | Allows you to search and report on reviews.  
The dropdown menu has links to recently visited reviews, as well as links to the Crucible Inbox and Outbox.  
Choose Reviews > Review dashboard to see the Review Dashboard that has more information about your reviews. | All screens  
All screens for logged-in users. |

Recent activity

The dashboard has an activity stream that displays recent commit activity and reviews activity. The activity stream will display your own activity as well as information from projects, reviews, people, repositories, etc, that you have selected as favourites. For more information on favourites, see Using favourites.

**Browsing commit activity**

Commit 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.

**Browsing reviews activity**

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).

Related pages

- Browsing source files
- Browsing Projects
- Viewing People's Statistics in Crucible
- Viewing reports
- Searching Crucible
- Using RSS feeds in Crucible
- Changing your User Profile

**Using the Review Dashboard**

Choose Reviews > Review dashboard from the header to see more information about your reviews.

*Screenshot: The Review Dashboard*
Using the 'My Reviews' sidebar

The 'My Reviews' sidebar contains information about reviews and snippets that you are involved with, for example how many reviews need your approval. Read more about the sidebar in Browsing all reviews.

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

**Inbox**

Click Inbox to show all reviews in these states:

**To Review** — Click to show all reviews where you are a reviewer and haven't yet completed your review work.

**Ready to Close** — Click to show all reviews where you are a moderator and haven't yet summarised and closed the review.

**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.

**Require My Approval** — Click to show all reviews where you are a moderator and need to approve the review.
Outbox

Click **Outbox** to show all reviews in these 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 **Archive** to show all reviews in these 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.

Related topics

- Browsing all reviews
- Using favourites
- 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 filter reviews by status, e.g. Open, Closed.

### To browse all reviews:

1. Choose **Reviews > Review dashboard**. The ‘Reviews’ page will display all open reviews, unless you have previously used a custom filter. See the screenshot below.
2. Browse your reviews, by clicking the links under ‘My Reviews’ and ‘My Snippets’ 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:

<table>
<thead>
<tr>
<th>All Open Reviews</th>
<th>Reviews that have not been summarised and closed yet.</th>
</tr>
</thead>
<tbody>
<tr>
<td>All Closed Reviews</td>
<td>Reviews that have been summarised and closed.</td>
</tr>
<tr>
<td>All Reviews</td>
<td>All reviews, including open reviews, closed reviews and draft reviews.</td>
</tr>
<tr>
<td>All Open Snippets</td>
<td>All open snippets.</td>
</tr>
<tr>
<td>All Snippets</td>
<td>All snippets, i.e. open and closed snippets.</td>
</tr>
</tbody>
</table>

*Screenshot: Browsing all open reviews*
Related Topics

Using the Review Dashboard

Viewing reports

Browsing source files

When FishEye is installed with Crucible, the **Repositories** tab is available in the header.

**To browse source files:**

1. Choose **Repositories > All repositories** from the header. 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 name for a 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. See the ‘Viewing a file’ screenshot below.
4. You can view various information about the file:

<table>
<thead>
<tr>
<th>Tab</th>
<th>Description</th>
</tr>
</thead>
</table>
| **Activity** | Shows recent activity for the item. There are a number of sub-options here:  
  - **All** — The default view, showing commits, reviews and JIRA issues.  
  - **Commits** — Shows commits in the activity stream.  
  - **Reviews** — Shows review activity in the activity stream.  
  - **Filter commits** — Applies constraints to the current activity stream.  
  - **Expand all** — Shows more detail for all changesets.  
  - **Scroll to changeset** — Displays the changeset ID specified |
| **Revisions**| When viewing a file, shows the latest revisions of the file. |
| **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. |
To download files, click the **Source** tab for the desired file, then right-click **Raw**.

**Screenshot: Viewing all repositories**

<table>
<thead>
<tr>
<th>Repository</th>
<th>State</th>
<th>Commit History (12 Months)</th>
<th>LoC</th>
<th>Commits</th>
<th>Reviews</th>
<th>Last Updated</th>
</tr>
</thead>
<tbody>
<tr>
<td>activeobjects</td>
<td>Stopped</td>
<td>-</td>
<td>-</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>applinks</td>
<td>Stopped</td>
<td>-</td>
<td>-</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>applinks-gt</td>
<td>Running</td>
<td>-</td>
<td>2,883</td>
<td>3</td>
<td>a few seconds ago</td>
<td></td>
</tr>
<tr>
<td>assassin</td>
<td>Running</td>
<td>22,433</td>
<td>546</td>
<td>12</td>
<td>a few seconds ago</td>
<td></td>
</tr>
<tr>
<td>assassin-http</td>
<td>Running</td>
<td>-</td>
<td>91</td>
<td>4</td>
<td>a few seconds ago</td>
<td></td>
</tr>
</tbody>
</table>

**Screenshot: Viewing a repository**

**Screenshot: Viewing a file**

### 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><img src="image1" alt="Icon" /></td>
<td>View review-level comments</td>
</tr>
<tr>
<td><img src="image2" alt="Icon" /></td>
<td>Go to the previous comment</td>
</tr>
<tr>
<td><img src="image3" alt="Icon" /></td>
<td>Go to the next comment</td>
</tr>
<tr>
<td><img src="image4" alt="Icon" /></td>
<td>Add a comment</td>
</tr>
<tr>
<td><img src="image5" alt="Icon" /></td>
<td>Go to the previous file in this review</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
- Related pages

#### 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 terms in the search box in the Crucible header. Crucible offers a number of criteria that you can use to refine your results, see [Refining your Quick Search Criteria](#) below.
2. Results will appear in a dropdown, as you type. Results will attempt to be matched against the review name, project and user.
   - If you want to use a quick search result, use the up- and down-arrows on your keyboard and press enter or use your mouse to select the item.
   - If the quick search 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 [Filtering Quick Search Results](#) below.
   Results are sorted by relevance and boosted if they were edited recently. A maximum of 10 results are displayed per page.
   - If you have [integrated your Crucible instance with a JIRA instance](#), you can display a summary of any JIRA issues referenced in your search results by hovering over the issue key. For more details, see [JIRA integration in Crucible](#).
4. If you want to run another search, enter your new criteria in the main search box or in the search box in the header.
   Note, only the search box in the header provides "quick nav" results.

**Screenshot: Quick Search displaying “quick nav” matches**
Refining your Quick Search Criteria

You can 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><strong>Field Handles</strong></td>
<td>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.</td>
<td>Enter &quot;CR-2818&quot; 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-2818.</td>
</tr>
<tr>
<td><strong>Searching for Discrete Strings</strong></td>
<td>Enter a specific string within quotation marks and Crucible will match against the exact string. Note, this search is not case-sensitive.</td>
<td>Enter &quot;CR-2818&quot; 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-2818.</td>
</tr>
</tbody>
</table>

Filtering Quick Search Results

You can filter Quick Search results using the controls in the left panel of the Quick Search page:

<table>
<thead>
<tr>
<th>Filter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>All repositories</td>
<td>Type a repository name into the field, or click the down arrow to see a list of repositories.</td>
</tr>
</tbody>
</table>
### Source type

<table>
<thead>
<tr>
<th>Source type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Click <strong>Files and directories, Commit messages, Diffs, Content or Committers</strong> to restrict the search results to just that source type.</td>
<td></td>
</tr>
</tbody>
</table>

### All Projects

<table>
<thead>
<tr>
<th>Description</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<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.</td>
<td></td>
</tr>
<tr>
<td><strong>Important:</strong> If you are using Fisheye with Crucible, there will be a repositories dropdown in the <strong>Source</strong> 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 <a href="#">Searching FishEye</a>.</td>
<td></td>
</tr>
</tbody>
</table>

### Reviews

<table>
<thead>
<tr>
<th>Description</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<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>
<td></td>
</tr>
</tbody>
</table>

### Comments

<table>
<thead>
<tr>
<th>Description</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Click this link to restrict your results to reviews that have comments that match the search criteria.</td>
<td></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><strong>Title</strong></td>
<td>Find reviews by searching for words within the title.</td>
</tr>
<tr>
<td><strong>Project</strong></td>
<td>Find reviews under a particular project.</td>
</tr>
<tr>
<td><strong>Author</strong></td>
<td>Find reviews moderated by a particular authors.</td>
</tr>
<tr>
<td><strong>Moderator</strong></td>
<td>Find reviews moderated by a particular moderator s.</td>
</tr>
<tr>
<td><strong>Creator</strong></td>
<td>Find reviews created by a particular creator.</td>
</tr>
<tr>
<td><strong>Reviewer</strong></td>
<td>Find reviews that are reviewed by a particular reviewer. This will default to the user logged in.</td>
</tr>
<tr>
<td><strong>Reviewer Status</strong></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><strong>Match Roles</strong></td>
<td>To use all the above filters, choose 'all'. To use any of the filters, choose 'any'.</td>
</tr>
<tr>
<td><strong>Review state checkboxes</strong></td>
<td>Check any of the review state checkboxes (e.g. 'Draft', 'Pending Approval') to filter for reviews in those states.</td>
</tr>
</tbody>
</table>

---

**Searching for Review Comments**
To search for review comments:

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

<table>
<thead>
<tr>
<th>Project</th>
<th>Find comments on reviews under a particular project.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Comment content</td>
<td>Find comments that contains the specified text.</td>
</tr>
<tr>
<td>Review Permalink</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>
<tr>
<td>Comment Author</td>
<td>Find comments made by a particular user.</td>
</tr>
<tr>
<td>Search Type</td>
<td>You can:</td>
</tr>
</tbody>
</table>

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

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

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 of 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 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. You can add your own avatar by uploading an image to an external service such as Gravatar, which Crucible supports. See Changing your User Profile.

### Screenshot: List of all People in Crucible (when using FishEye with Crucible)
Viewing a Person’s Activity Screen

Click on a username to see a listing of activity for them as well as charts showing statistics for their activity.

The right hand pane displays a list of all activity for 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
- click the star icon to add an item to your favourites.

The left hand pane displays charts around this activity, including:

- number of active reviews
- charted history of lines of code
- code committing activity
- general statistics.

Screenshot: The People Activity Screen in Crucible
Viewing charts of 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

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.

- About
  The username section shows the email address, then the first and latest commit dates for the person in context. Also displayed are data points for the previous week and all-time. It shows number of commits, number of files changed and number of lines changed.
Reviews
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, Ready to Close, Out For Review, Open and Closed.

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

Commit Activity
The Commit Activity section shows four 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; the last shows a commit calendar.

Committer Mappings
The Committer Mappings section displays username mappings from various systems if they have several usernames in play.
### Commit Activity

52 week commits volume

Commits by day

Commits by hour

Commit calendar

Username jxie is mapped to:
- jxie in FE
- jxie <jxie@atlassian.com> in FE-hg
- jxie <jxie@atlassian.com> in FE-hg
- jxie <jxie@atlassian.com> in FE-hg
- jxie <jxie@atlassian.com> in atomicreptile
- jxie in FEP
- jxie <jxie@atlassian.com> in atlassian-http

### Browsing Projects

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

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

<table>
<thead>
<tr>
<th>Sub-Tab Name</th>
<th>Description</th>
</tr>
</thead>
</table>
| **Activity** | • All Activity — The default view.  
• Commits — Shows commits in the project (visible when using FishEye).  
• Reviews — Shows reviews in the project.  
• Issues — Shows JIRA issues related to this project.  
• 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. |

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

**Screenshot: The Crucible Projects Index**

<table>
<thead>
<tr>
<th>Projects</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project</td>
</tr>
<tr>
<td>Default Project</td>
</tr>
<tr>
<td>Build Monitor</td>
</tr>
<tr>
<td>Clover Eclipse Plugin</td>
</tr>
<tr>
<td>Confluence Hosted</td>
</tr>
<tr>
<td>Clover IDEA plugin</td>
</tr>
<tr>
<td>Clover</td>
</tr>
<tr>
<td>FishEye</td>
</tr>
<tr>
<td>JIRA FishEye Plugin</td>
</tr>
<tr>
<td>Jira Studio</td>
</tr>
<tr>
<td>LinkChecker</td>
</tr>
<tr>
<td>Fournalls</td>
</tr>
</tbody>
</table>

**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](chart1.png)

**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>
</table>

Created by Atlassian in 2013. Licensed under a [Creative Commons Attribution 2.5 Australia License](http://creativecommons.org/licenses/by/2.5/au/).
52 week commits volume
This chart shows the amount of commits, shown by week over a one year period.

Commits by day
This chart shows the amount of commits, graphed by day over the past week.

Commits by hour
This chart shows the amount of commits, graphed by hours over the past day.

Commit calendar
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.

Changing your User Profile

See Changing your User Profile in the FishEye documentation.

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

<table>
<thead>
<tr>
<th>You have 13 favourites.</th>
</tr>
</thead>
<tbody>
<tr>
<td>🌟 FE (SVN repository)</td>
</tr>
<tr>
<td>🌟 Clover Eclipse Plugin</td>
</tr>
<tr>
<td>🌟 Confluence Hosted</td>
</tr>
<tr>
<td>🌟 Clover IDEA plugin</td>
</tr>
<tr>
<td>🌟 JIRA FishEye Plugin</td>
</tr>
<tr>
<td>🌟 Jira Studio</td>
</tr>
<tr>
<td>🌟 LinkChecker</td>
</tr>
<tr>
<td>🌟 A Good Marketing ploy</td>
</tr>
<tr>
<td>🌟 Build Monitor</td>
</tr>
<tr>
<td>🌟 Default Project</td>
</tr>
<tr>
<td>🌟 Clover</td>
</tr>
<tr>
<td>🌟 Testing Project</td>
</tr>
<tr>
<td>🌟 FishEye</td>
</tr>
</tbody>
</table>

Renaming a Favourite

To rename a favourite,

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

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

3. Enter the new name for the favourite in the 'Name' field and click the 'Save label' button. The favourite will be renamed on the dashboard.

Screenshot: The 'Update favourite' dialogue
Removing a Favourite

To view your favourites,

1. Click the favourites icon (⭐) in the header.
   The code reviews, changesets, files, people and repositories that you have selected will be displayed in a list on the dashboard (see screenshot).
2. Click the favourites icon (⭐) next to a favourite to show the 'Update favourite' dialogue (see screenshot below).
3. Click the 'Remove' button. The code review, changeset, file, person or repository will be removed as your favourite (the icon will be greyed out) and will not display the next time you view your list of favourites.

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

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
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.

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.

**Screenshot: Adding a 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.

**Screenshot: Adding a Changeset to Your Favourites**

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
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.

Using Keyboard Shortcuts in Crucible

To see the available shortcuts, navigate to a review in Crucible, then choose Tools > Keyboard Shortcuts.

General shortcuts

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

Custom navigation

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

Comment navigation

<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>
</tbody>
</table>
### File navigation

<table>
<thead>
<tr>
<th>Key</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>j</td>
<td>Go to the next element (file, comment, defect or diff hunk depending on your current context)</td>
</tr>
<tr>
<td>k</td>
<td>Go to the previous element (file, comment, defect or diff hunk depending on your current context)</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 previous unreviewed file</td>
</tr>
<tr>
<td>i</td>
<td>Go to next unreviewed file</td>
</tr>
<tr>
<td>y</td>
<td>Set file reviewed and go to next unreviewed file</td>
</tr>
<tr>
<td>shift + y</td>
<td>Toggle file reviewed/unreviewed status</td>
</tr>
<tr>
<td>e</td>
<td>Expand current file</td>
</tr>
<tr>
<td>c</td>
<td>Collapse current file</td>
</tr>
<tr>
<td>shift + e</td>
<td>Expand all files</td>
</tr>
<tr>
<td>shift + c</td>
<td>Collapse all files</td>
</tr>
</tbody>
</table>

### The Crucible workflow

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

On this page:

**Crucible Workflow**

**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

**Roles**

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:

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

**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.

Diagram: Workflow for One-to-One Reviews

![Workflow Diagram](Diagram: Workflow for One-to-One Reviews)

Need more information? Read more about the different forms of workflow in Crucible.

Next, we explore an example workflow for a two-person code review in Crucible.

Example workflow: Two participant code review

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

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

![Screenshot](Screenshot: Opening a review from the FishEye Source view)

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

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

*Screenshot: Creating a review in the Edit Review Details dialogue*

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

The author can also add more content to the review, if they wish, by clicking **Add Content**. See **Adding content to the review**.

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

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

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

### 2. The reviewer comments on the code

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

**Screenshot: A Crucible review notification email**

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

**Screenshot: The Crucible Review screen**
The Review screen displays the source files that are under review. The reviewer clicks file names to see the code to be reviewed. As the reviewer reads the changes, they can add comments:

- Click **Add a general comment** (under 'General Comments' on the Review screen) to comment on the overall review.
- Click **Add a file comment** (just above the source code listing) to add a general comment about a source code file.
- Click on any line in the source file to enter a comment there (multiple lines can be selected by clicking and dragging).

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

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

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

### 3. The author responds to the comments

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

**Screenshot: Comment threads in Crucible**
4. The author closes the review

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

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

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

Screenshot: Viewing a closed review

If the author ever needs to resume work on the closed review, they can simply click Reopen when viewing this screen. This returns the status of the review to "Open", without changing the status of existing reviewers. Click Edit Details to add reviewers or to change other details of the review.

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 in 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*.

**Roles and Status Classifications**

This page explains the roles and 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 *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.

**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.

**Creating a review**

This page provides an overview of how to create a review in Crucible.

More information about the stages in creating a Crucible review can be found on these pages:

- Add content to the review
- Choose reviewers
- Perform the review
- Summarise and close the review

You can also create Crucible reviews in these other ways:

- Creating a review from FishEye
- Creating a review from JIRA
- Creating a review from a URL
- Creating a Snippet Review
- Creating reviews from the command line

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

**On this page:**

- Create a new review
- Add content to the review
- Edit the review details
- Considerations when creating reviews

**Create a new review**

Within Crucible, create a new review by clicking Create review in the header. Select the project for the review (if you have multiple projects), and click Create Review.
Add content to the review

In the ‘Add Content to Review’ dialog, click one of the options to choose files or changesets for review. Crucible supports post-commit and pre-commit review types – these simply depend on the type of content you add to the review:

- **Post-commit reviews** – You want code changes that have already been committed to the SCM to be reviewed. In Crucible, you can browse and search for files or changesets in the SCM, which you then add to the review. Crucible can suggest files that should be considered for review, based on recent activity.
- **Pre-commit reviews** – You want code changes that have not yet been committed to the SCM to be reviewed. You need to create a patch file for the code changes and then, in Crucible, add the patch to the review.

See [Adding content to the review](#) for more details.

- **Browse Changesets** – browse for SCM changesets
- **Explore Repositories** – browse for SCM files
- **Search for Files** – or changesets, in the SCM
- **Suggest Files** – get Crucible to suggest files
- **Pre-commit** – upload patch files
- **Attachments** – upload any type of file

Note that Crucible supports iterative reviews – for both post-commit and pre-commit reviews you can update the review with new versions of files, and changesets, created after the review was started. Crucible allows the reviewer to see the different versions of updated files, so as to understand the changes that have been made.
Click **Edit Details** to move to the next stage.

**Edit the review details**

Here you can choose a title, reviewers, objectives, due date, linked reviews and issues:

Once you're finished, click **Done**.

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.

When the reviewers have performed their reviews, you can summarise and close the review.
Considerations when creating reviews

Review effectiveness

Based on our own experience of over 13000 reviews, we have found that reviews with fewer files and reviewers are more effective. We have seen effects such as:

- Time spent reviewing each file decreases as the number of files increases.
- Reviewers spend less time reviewing as the number of reviewers increases.
- Reviewers find fewer defects as the number of reviewers increases.

This suggests that reviews should be created with care to get the best value from them:

- Avoid overloading the review. Reviews should be focussed on just a few necessary files.
- Avoid overcrowding the review. Reviewers should be selected with care, and should be guided individually on what to look for.

Crucible performance

The performance of a Crucible instance can be seriously degraded if very large reviews are created.

To prevent a user from accidentally causing this, Crucible has a limit on the review content size when creating reviews. The limit is 800 file revisions.

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 and not is deleted", or similar logic.

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

Creating a review from FishEye

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

On this page:

1. Open the FishEye Source view
2. Create 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. Create the review

To create the review:

- From the FishEye Source view, choose Reviews > Create Review.
3. Choose a project

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

Creating a review from JIRA

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

JIRA must be integrated with both FishEye and Crucible before you can do this. See Linking Crucible to JIRA for information on how to set this up.

Note that FishEye content appears on the Source tab, and Crucible content on the Reviews tab, in JIRA.

See also Creating JIRA issues from the review.

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 Create review to create a new review for a particular changeset.
   b. Click Create review for all commits to include all changesets from the JIRA issue in the new review.
4. If a similar review already exists, you can add the changesets to that.
5. In Crucible, the new review is in edit mode:
   - The content of the changeset becomes the content (i.e. files) to be reviewed.
   - The author of the changeset becomes the author of the review, if Crucible is aware of this user. Otherwise the creator of the review becomes the author.
   - The creator of the review becomes the moderator.
   - The commit log message is used as both the Title and Statement of Objective.

6. Choose Tools > Start Review, in Crucible, when you are ready.

The next step is to add reviewers.

Screenshot: Adding a review from within JIRA
Creating a review from a URL

You can set up a URL that 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/csids (where &quot;%2F&quot; is the URL-encoded form of is '/')</td>
<td>Yes</td>
</tr>
<tr>
<td>repo</td>
<td>The name of your repository.</td>
<td>Yes (unless supplied in the csid)</td>
</tr>
<tr>
<td>title</td>
<td>The title of your new Crucible review.</td>
<td>No</td>
</tr>
<tr>
<td>description</td>
<td>The description of your new Crucible review.</td>
<td>No</td>
</tr>
</tbody>
</table>

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

- The content of the changeset becomes the content (i.e. files) to be reviewed.
- The author of the changeset becomes the author of the review, if Crucible is aware of this user. Otherwise the creator of the review becomes the author.
- The creator of the review becomes the moderator.
- 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 from the Create review menu 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.

You can see your own snippets, or everyone's snippets, by choosing Reviews > Review dashboard. See Searching Crucible for information about filtering snippet reviews.

Screenshots: Creating a Snippet Review (click to view larger images)

Creating reviews from the command line

You can use the Review CLI tool to create reviews in Crucible, for patches and commits, directly from your terminal. It takes the uncommitted changes in your workspace and creates a review for them in Crucible. The Review CLI tool may be especially useful if:

- you often create pre-commit reviews
- you want to submit a diff from an external tool for review

You can use the tool on Windows, Linux and Mac OS X, for repositories that are managed in:

- Subversion
- Perforce
- Git
- Mercurial
- CVS

The tool supports Crucible 3.0, and later versions. Python 2.7 must be installed on your local machine (Python 3 is not supported).

You can use the tool to create new reviews, and to update existing reviews with new patches.

Known limitations

- The CLI tool does not yet allow you to create a review for an existing commit.
- The CLI tool takes all the files that have been modified, added or removed in the working copy and submits those for review. It doesn't support choosing only files associated with a specific changeset.
- You can only use the CLI tool to create reviews in projects for which you already have review create permission.
Installing the Review CLI tool

Download the Review CLI tool from either of the following locations in Crucible:

- When creating a review, click **Pre-commit** in the 'Add Content to Review' dialog, then click **Download**:

![Add Content to Review CR-FE-7780](image)

- Choose **Profile settings** from your user menu, and then **Tools**. Click **Download**:

![Profile settings](image)

The script comes pre-configured for the user for the given SCM server.

Copy the script to a location somewhere in your system path, for greatest ease of use.

**Python**

Python 2.7 is required.

**Linux**

Python 2.7 should come with your distribution. If not, or if 2.7 isn’t installed, you will need to install the appropriate package. Please refer to your package manager for the appropriate version.

**Windows**
You can install Python by running the latest Python 2.7 Installer for Windows.

**MacOS X**

Python 2.7 should come with any modern version of MacOS X. You can also download Python 2.7 from [http://www.python.org/getit/](http://www.python.org/getit/).

Using the Review CLI tool

To create a review using the Review CLI tool, run the script from a directory under SCM control that has local, uncommitted changes. The changes will be submitted to Crucible for review.

Call the script with the following command:

```
crucible.py <arguments list>
```

The Review CLI syntax is consistent with that for Crucible and FishEye smart commits. Examples of syntax usage are provided in the following table:

<table>
<thead>
<tr>
<th>Action</th>
<th>Syntax</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>View usage help</td>
<td><code>crucible.py --help</code></td>
<td>Displays the help with descriptions of all the valid commandline arguments</td>
</tr>
<tr>
<td>Create a pre-commit review</td>
<td><code>crucible.py</code></td>
<td>Gets the patch from the current SCM, prompts for the project and review title, and creates a draft review</td>
</tr>
<tr>
<td>Create a pre-commit review</td>
<td><code>crucible.py -m &quot;the review title&quot; CR-FE</code></td>
<td>Gets the patch from the current SCM, creates a draft review in the CR-FE project with the given title</td>
</tr>
<tr>
<td>Create a pre-commit review</td>
<td><code>crucible.py CR-FE @matt @joe --moderator ted</code></td>
<td>Gets the patch from the current SCM, creates a review in the CR-FE project, adds matt and joe as the reviewers and ted as a moderator, starts the review</td>
</tr>
<tr>
<td>Create a pre-commit review</td>
<td><code>crucible.py -r repo1</code></td>
<td>Creates a pre-commit review interactively, trying to anchor the patch to the given repository</td>
</tr>
<tr>
<td>Add a patch to a review</td>
<td><code>crucible.py CR-FE-1204</code></td>
<td>Gets the patch from the current SCM and adds it to the review CR-FE-1204</td>
</tr>
<tr>
<td>Create a review from diff output</td>
<td>`hg diff</td>
<td>crucible.py`</td>
</tr>
<tr>
<td>Create a review from a diff file</td>
<td><code>crucible.py -f file.diff</code></td>
<td>Creates a pre-commit review interactively, taking the output of the patch from the given file</td>
</tr>
</tbody>
</table>

**Adding content to the review**

This page explains how to add content, such as files and changesets, to a Crucible review. The procedure depends on whether you are setting up a post-commit or pre-commit review:

- **Post-commit reviews** – You want code changes that have already been committed to the SCM to be reviewed. In Crucible, you can browse and search for files or changesets in the SCM, which you then add to the review. Crucible can suggest files that should be considered for review, based on recent activity.
- **Pre-commit reviews** – You want code changes that have not yet been committed to the SCM to be reviewed.
In Crucible, you need to **add patch files to a pre-commit review.**

### Related pages:
- Creating a review
- Iterative reviews
- Choosing reviewers
- Performing the review

Note that Crucible supports 'iterative' reviews – for both post-commit and pre-commit reviews you can update the review with new versions of files, and changesets, created after the review was started.

### On this page:
- General procedure
- Selecting changesets for review
- Selecting repository files for review
- Searching for files to review
- Adding patch files to a pre-commit review
- Adding attachments to a review
- Choosing the way files are added to the review

#### General procedure

**To add content to a review:**

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

2. In the 'Add Content to Review' dialog, click an option for how you wish to add content to the review, and then follow the instructions in the relevant section below:

#### Post-commit reviews

<table>
<thead>
<tr>
<th>Browser Changesets</th>
<th>Allows you to choose changesets from a Source Code Management (SCM) repository. See Selecting changesets for review below.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Explore Repositories</td>
<td>Allows you to browse for files in a SCM repository. This option only appears when FishEye is installed. See Selecting repository files for review below.</td>
</tr>
<tr>
<td>Search for Files</td>
<td>Allows you to search a SCM repository for files or changesets. This option only appears when FishEye is installed. See Searching for files to review below.</td>
</tr>
</tbody>
</table>

#### Pre-commit reviews

<table>
<thead>
<tr>
<th>Pre-commit</th>
<th>Allows you to upload patch files to a review. See Adding patch files to a pre-commit review below.</th>
</tr>
</thead>
</table>

| All reviews | Allows you to upload any file to a review, including binary files and files outside of a SCM repository. See Adding attachments to a review below. |

3. Click **Done** to save your changes, without starting the review.

#### Selecting changesets for review

Click **Browse Changesets** in the 'Add Content to Review' dialog to add SCM changesets to your review.
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 dialog:

<table>
<thead>
<tr>
<th>Repository</th>
<th>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.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Author</td>
<td>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.</td>
</tr>
<tr>
<td>Branch</td>
<td>This will only show files and recent changes on that branch from the repository set above.</td>
</tr>
<tr>
<td>Commit message</td>
<td>Filter for specific commit messages.</td>
</tr>
<tr>
<td>Add to Review as</td>
<td>Choose the form of the review. See Choosing the way files are added to the review below.</td>
</tr>
<tr>
<td>Scroll to changeset</td>
<td>Allows you to jump to a particular change set by entering its title and pressing Enter.</td>
</tr>
</tbody>
</table>

Select the checkbox next to a changeset ID to add the entire changeset. Note that:

- 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?

Selecting repository files for review

Click Explore Repositories on the ‘Add Content to Review’ dialog to browse the SCM repositories for files to add to your review:

- By default, the folders are sorted by path name but they can also be sorted by last-commit or first-commit.
- To select a particular revision of a file, select Load full history... from the revision number list. This will refresh the available options in the list.

Note that:

- Empty folders are greyed out.
- The ‘Cog’ menu has options to Hide empty directories and to Hide deleted files.
Searching for files to review

Click **Search for Files** on the 'Add Content to Review' dialog to search for files to add to your review.

Search is only available when using **FishEye** with Crucible.

Adjust the search filters to find the files you need. If the simple filters are not enough, consider using **EyeQL** queries.

Read more about searching your repositories in the FishEye documentation.

Adding patch files to a pre-commit review

Click **Pre-commit** on the 'Add Content to Review' dialog to add previously created patch files to a pre-commit review.

Choose an upload method:

- **Select file from the file system**
  - **Charset** – click the edit icon (✍️) to choose the character set being used. **UTF-8** is the default.
  - **Choose File** – click to to browse for the file that you want to add to the review.

- **Paste text from clipboard**
  - **Patch text** — paste your copied text in this text area.
For more information see Creating patch files for pre-commit reviews.

**Add Content to Review TEST-400**

**Upload patch manually for pre-commit review**

- **Upload methods**
  - Select file from file system (max file size: 10MB)
  - Paste text from clipboard

- **Charset** UTF-8
- **File** Choose File
  - No file chosen
  - **Upload**

**Patch anchoring**

A short-coming of patches for code review is the reduced context around code changes because the patch does not include all lines of code from the file. Crucible ‘patch anchoring’ overcomes this by searching for the relevant file content in the connected repositories, and automatically anchoring the patch to the trunk or the branch with the most recent commit activity. Crucible can then seamlessly display more context, as required.

See Using Crucible patch anchoring for more information.

**Adding attachments to a review**

Click **Attachments** on the ‘Add Content to Review’ dialog to add attachments to your review.

You can attach additional files to be used in the review, including binary files, images or code files that are not stored in a version control repository.

| **Charset** | Click to choose the character set being used. UTF-8 is the default. |
| **Choose File** | Browse for a file that you want to add to the review. |
| **Upload** | Browse for a file that you want to use as the base of a diff with a previously attached file. |

To add another iteration of a file, upload a different version of the file with the same filename. It will be added as a new version.

**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 list:
Iterative reviews

Crucible supports iterative (cumulative, or incremental) reviews for both post-commit and pre-commit reviews. This allows you to update the review with new versions of files, and changesets (for post-commit reviews) or patches (for pre-commit reviews) that have been created after the review was started.

Iterative reviews allow your team to discuss changing code in the context of a single review. This lets the reviewers see all the related changes together, and to more easily keep track of comments and defects.

On this page:
- Iterative post-commit reviews
- Iterative pre-commit reviews
  - Initial patch upload
  - Iterative patch uploads
- Viewing diffs

Iterative post-commit reviews

To set up an iterative post-commit review, you create a review, and add content to it, in the usual way. Crucible automatically recognises when files under review have been updated in the repository, and provides the option to add the revision to the review.

Iterative pre-commit reviews

Pre-commit reviews make use of patch files that are uploaded to a review. Crucible allows revisions of patch files to be uploaded to the review, and can display diffs for files in the patches. This allows your team to set up...
and perform iterative pre-commit reviews.

See Creating patch files for pre-commit reviews.

*Initial patch upload*

When uploading the initial patch for a review, Crucible must be able to anchor the patch to a repository if you subsequently want to upload patch iterations. If Crucible is unable to anchor the patch to a repository, you will only be able to upload the patches as separate files.

You upload the initial patch for a review in the usual way — see Adding content to the review.

*Iterative patch uploads*

When you add a new iteration of the patch to the review, you can choose which previously uploaded patch it is a revision of. The new patch must be anchored to the same repository as an existing patch.

Note that you cannot add unanchored patches, even if they include full-context diffs. You can include an unanchored file in the anchored patch, however Crucible is unable to provide full context for that.

---

**Add Content to Review CR-10**

**Upload patch manually for pre-commit review**

Upload methods
- Select file from file system (max file size: 10MB)
- Paste text from clipboard

Charset: UTF-8

File: Choose File

**Existing patches in CR-10**

CR-10-patch-0.txt (6 minutes ago) - 2 iterations

- **CR-10-patch-0.txt**: (anchored to svnsample1 : / )
  - trunk/file
  - trunk/copied-and-modified
  - trunk/to-be-modified
  - trunk/moved-and-modified
  - trunk/to-be-moved-and-modified
  - trunk/copied-and-modifie2
  - trunk/added
  - trunk/to-be-removed
  - Mainline/foo

- **CR-10-patch-1.txt**: (anchored to svnsample1 : / )

**Viewing diffs**

Crucible allows the reviewer to see the different revisions of a file within the same review. The ‘slider’ in the file
view allows you to interactively select which revisions are compared in the displayed diff, and to see the full source content. Comments are connected to, and displayed with, a specific revision. This allows you to review every change that has occurred on a code file over a given range of commits, and lets you see the evolution of the file through various revisions (all within one Crucible review).

These screenshots show how, for a post-commit review, you can drag the slider 'handles' so as to display just the changes in the last commit:

![Screenshot of Crucible displaying diff](image)

Drag the 'handles' to the same commit to see the full source of that version of the file.

When viewing patch files in a pre-commit review, the slider displays the diff for the selected iterations, in a similar way to that for post-commit reviews. Each patch iteration is referred to as a 'working copy'.

![Screenshot of Crucible displaying patch](image)

Creating patch files for pre-commit reviews

This page describes how to create patch files from your local repository, how to attach them to a Crucible review and how to use Crucible's Patch Anchoring to retrieve more context from the original file.

A patch file is a portion of a source code file that contains the code changes that you have made – it's a diff that shows the differences between your working copy and the base revision.

A pre-commit review in Crucible allows a developer's code changes, in the form of a patch file, to be reviewed before those changes are committed to the SCM. A typical scenario is where the developer does not have write access to the repository. The developer creates the patch file and adds it to a Crucible review. Once reviewed, the patch is either committed to the repository or is sent back to the author.

You can create the patch file from your local repository:

- using tools in your IDE – described below
- using repository command-line tools
- using the Crucible Review CLI tool – see Creating reviews from the command line

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

As an enhancement, Crucible's patch anchoring adds context:

- By default, patch files will only show a few lines of code surrounding each change, rather than the entire file and its changes. This feature overcomes this limitation.
- Note however that anchoring is done as an afterthought, and can be reverted.

On this page:

1. Select a parent folder, sub-folder or file that you have altered, in the Project tool window.
2. Select Version Control > Create Patch:
3. Click **Create Patch**, choose a location to save the patch file to, and click **OK**.

   If you do not have the Create Patch command available in 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**.

2. In the Local History view, right-click the revision number, and choose **Create Patch**.
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

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

2. In the Create Patch window, choose a location on your computer and type an appropriate file name (the file format is plain text):
Creating a patch file from the command line

Use the following commands to create patch files from the respective SCMs. `patch.txt` represents your name for the new patch file.

<table>
<thead>
<tr>
<th>SCM</th>
<th>Command</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>CVS</td>
<td><code>cvs diff -Nu &gt; patch.txt</code></td>
<td>Creates a patch file with around three lines of code, before and after each change. Revision information is included by default, so if you have FishEye, you may be able to anchor the patch to get full context.</td>
</tr>
<tr>
<td></td>
<td><code>cvs diff -N -U 10000 &gt; patch.txt</code></td>
<td>Creates a patch file that shows all code in the file. 10000 refers to the number of code lines before and after each change that are included in the patch.</td>
</tr>
<tr>
<td></td>
<td>Command</td>
<td>Description</td>
</tr>
<tr>
<td>------</td>
<td>----------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>SVN</td>
<td><code>svn diff &gt; patch.txt</code></td>
<td>Creates a patch file with around three lines of code, before and after each change. Revision information is included by default, so if you have FishEye, you may be able to anchor the patch to get full context.</td>
</tr>
<tr>
<td></td>
<td><code>svn diff --diff-cmd diff -x &quot;-U 10000&quot; &gt; patch.txt</code></td>
<td>Creates a patch file that shows all code in the file.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- The built-in diff feature in <code>svn diff</code> does not support specifying lines of context, so you must tell Subversion to use an external diff command.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- The second <code>diff</code> in the command needs to be the name of your external diff command. You might need to specify the full path to that command, such as <code>/usr/bin/diff</code>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- On the Windows platform, you may need a Unix-like emulator such as Cygwin, and install the optional diff command for that.</td>
</tr>
<tr>
<td>Perforce</td>
<td><code>p4 diff -dcu &gt; patch.txt</code></td>
<td>Creates a patch file with around three lines of code, before and after each change. Revision information is included by default, so if you have FishEye, you may be able to anchor the patch to get full context. The <code>-dcu</code> option provides a combination of &quot;context format&quot; and &quot;unified format&quot;. 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).</td>
</tr>
</tbody>
</table>
### Using Crucible patch anchoring to automatically add full context

Crucible's Patch Anchoring feature allows you to add a regular patch (showing only a few lines of context) to a review. 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 files so that you can view all of the lines of code (greatly enhancing the review process).

#### To use patch anchoring:

1. Click **Create review** in the Crucible header.
2. Click **Pre-commit**. In the dialog that appears, click **Choose File** to locate your file, then **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><strong>Version</strong></th>
<th><strong>Command</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Perforce</strong></td>
<td><code>diff -u /dev/null path_to_added_file &gt;&gt; patch.txt</code></td>
<td>Example of using GNU diff to append files individually to the patch in UNIX. Since Perforce diffs do not include added and deleted files, you should use <code>p4 opened</code> to find such files. Replace <code>path_to_added_file</code> with the actual path of your added file. You can follow a similar procedure with deleted files using <code>p4 print</code> to extract the previous version of the file.</td>
</tr>
<tr>
<td><strong>Mercurial</strong></td>
<td><code>hg diff &gt; patch.txt</code></td>
<td>Creates a patch file with around three lines of code, before and after each change. Revision information is included by default, so if you have FishEye, you may be able to anchor the patch to get full context. 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.</td>
</tr>
<tr>
<td><strong>Git</strong></td>
<td><code>git diff &gt; patch.txt</code></td>
<td>Creates a patch file with around three lines of code, before and after each change. Revision information is included by default, so if you have FishEye, you may be able to anchor the patch to get full context.</td>
</tr>
</tbody>
</table>
• 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.
• Start the review. When viewing the diffs, you will be able to choose more than three lines of context from the **View** menu.

**Screenshot: Crucible Patch Anchoring**

**Screenshot: Editing the settings for Patch Anchoring**
Choosing 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 and removing reviewers
  - Adding users to a review
  - Suggested reviewers
  - Inviting unregistered users to the review
  - Removing reviewers
- Checking the draft and starting the review
- Next steps

Entering basic information

Once a review has been created, the Edit Review dialog opens:
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 and removing 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 **Done** 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, select **Allow anyone to join**.

**Suggested 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 clicking.

**Inviting unregistered 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 **Public Signup** must be set to ON. Go to the Admin area and click on **Authentication**, under 'Security Settings', and look in the 'Built-in' section.
To invite an external user to a review:

1. Create a new review.
2. On the Create New Review screen, simply type the user's email address into the Reviewers field, then press Enter to select.
3. Click Save to save the draft review. The users are not sent any information at this time.
4. When you click Start Review, this is when all email invites and notifications are sent out.
5. The external user will receive an email address from the Crucible server, containing a special URL that they can visit.
6. 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.)
7. 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.

Removing reviewers

You can remove reviewers who no longer need to contribute to the review.

Click Edit Details to open the 'Edit Review Details' screen, hover over the reviewer's name, and click the cross.

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 **Starting a review**.

**Performing the review**

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

**On this page:**

- Browse your reviews on the Dashboard
- Browse all reviews on 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 on the Dashboard**

When you first start Crucible, the Dashboard is displayed, which shows your current reviews and other activity related to you.

Use the 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. The reviews then move to the **To Summarize** list.

Read more about using the **Dashboard**.

**Browse all reviews on the 'Reviews' tab**

All reviews that involve you in any role are listed when you click **Inbox** or **Outbox** under the **Reviews** menu. For example, choose **Reviews > Inbox** to see reviews that don't require further action from you, but are still active.

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, when viewing the file:

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

Next steps

- Starting a review
- Adding comments
- Changeset discussions
- Flagging defects
- Creating JIRA issues from the review
- Viewing reports
- Completing your review
- Sending a review's comments via Email
- Using the Review History Dialog
- Using RSS feeds in Crucible
- Using Wiki Markup in Crucible
- Tracking Crucible Review Metrics

Starting a review

On this page:

- Starting a review
- Editing a review once started

Starting a review

Starting 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 **Start Review**.
• If you are not the moderator of your review, click **Send to Moderator.** This changes the state to 'Requires 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

---

**i** Note that only people with the 'Approve' permission can start a review.

Editing a review once started

You can quickly add files to, or remove files from, the content of a review at any time:

- Add files to a review by clicking the 'Add content' button.
- Remove files by clicking the 'Edit' button in the left navigation panel, then clicking the red cross icons beside files to be removed.

You can edit all the details of a review, including the content, by clicking **Edit Details** near the top right of the review.

Screenshot: Launching Edit Mode

---

**Screenshot: Crucible Edit Mode for Review Content**
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 comments
- Draft comments

Locating existing comments

The number beside a filename, in the left-hand panel of a review, indicates the number of comments on that file. (The number of unread comments, if any, is shown in brackets.)
Adding comments

There are various types of comments that you can add in Crucible:

<table>
<thead>
<tr>
<th>To comment on...</th>
<th>Do this...</th>
</tr>
</thead>
<tbody>
<tr>
<td>The whole review</td>
<td>Click <strong>General Comments</strong> (in the left-hand panel), then <strong>Add a general comment</strong> (under 'General Comments').</td>
</tr>
</tbody>
</table>

---

**General Comments**

- Lukasz Piatek says:
  - Piggybacking ProjectListResource handler
  - Reply | Leave Unread | Close

- Piotr Święcicki says:
  - It's a pure pleasure to review your code!
  - Reply | Leave Unread | Close

---

Created by Atlassian in 2013. Licensed under a [Creative Commons Attribution 2.5 Australia License](https://creativecommons.org/licenses/by/2.5/au/).
A source file under review

Click on the file in the left-hand panel, then click **Add a file comment** (just above the source code listing).

Lines of code

Click on a line of code in the displayed source file of a review.

- You can click and drag to select multiple lines, and click individual lines to select or deselect them. The comment will appear in the source at the last line selected.
- Hover over the comment to see the lines to which the comment applies.
- To select text on the page without adding a comment, hold down the Alt key while dragging the cursor.

A revision or changeset

See **Changeset discussions**.

To reply to a comment, click **Reply** at the bottom of the comment.

⚠️ Only people with the 'Comment' permission can add comments. A comment can only be deleted by the author of the comment.

ℹ️ Read about flagging defects too.

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.

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.

ℹ️ You need to be logged in to create changeset comments.

Adding comments to changesets

To add a comment to a changeset:

1. Click on a changeset on the **Commits** tab for the repository. Display comments by clicking **Discuss** at the upper right corner, or the speech bubble icon 📣 in the left margin.
2. Click **Add a comment** (under the repository details near the top left).
3. Type your comment. If required, you can tag your comment as being a defect note by clicking **Defect**.
4. Click **Post**.

Once submitted, others can respond to your comment by clicking **Reply**. Replies are threaded as separate comment discussions. You can right-click on the permalink icon 👇 to copy a link to the comment. The comment author can edit or delete their own comments.

To hide the changeset comments, click the page icon 📊. You can display the comments panel 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 area:

1. In the Admin area, click Repositories (under 'Repository Settings' on the left).
2. Find your repository and choose View from the 'cog' menu in the Actions column.
3. Click Other Settings in the left panel.
4. Under 'Changeset Discussions' clear the Allow changeset discussions checkbox.

By default, changeset discussions are on.

Notifications

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

Flagging defects

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

To do this, simply check Defect when adding a comment and select a category from the drop-down list.

Screenshot: Defects

You may want to mark comments as defects in order to associate defect classifications, or simply to highlight to the author or moderator that the issue you raised in your comment requires attention. You can use the with defects filter to find files that have been flagged with defects.

- 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.
Creating JIRA issues from the review

From any review comment (general, file, inline) in Crucible, you can create a JIRA issue directly from the comment. This requires that Crucible is integrated with JIRA, version 5.0 or later, and is disabled if you have an earlier version of JIRA.

Inline issue creation allows:

- Tracking of the status of the comment
- A faster way to pull out incidental suggestions raised in reviews as JIRA issues
- A quick link back to the comment from the JIRA issue, using Remote Issue Links.

You might find this useful when:

1. Tracking the status of a review:
   a. The 'Issues Raised from Comments' section in the review shows the open/closed status of related issues.
   b. Raising related issues enforces dealing with subtasks before the review can be closed.
2. Closing off a review:
   a. You can create JIRA issues, unrelated to the current review, to track matters to be dealt with later.

Creating a JIRA issue

To create a JIRA issue from a review, click Create Issue in an existing comment. Note that you need the 'Comment' permission in Crucible to see the Create Issue link.

Crucible suggests a JIRA instance, project and issue type, but you can choose from the available options. You can choose Sub-task from the JIRA Issue Type list if a JIRA issue is already linked to the review.

Crucible only displays required fields for the issue type; these can be configured in JIRA by your administrator.

Once the issue is created, the comment displays a link to the issue in JIRA, and in JIRA, the issue displays a link back to the comment in Crucible. The 'Issues raised from comments' section of the review displays links to the JIRA issues.
See Creating a review from JIRA.

Viewing reports

This page describes 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'.

On this page:

- Viewing the Review Blockers report
- Viewing the JIRA Blockers report

See also:

- Viewing the Review Coverage report

Viewing the Review Blockers report

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

1. Click the dropdown arrow next to the Reviews tab at the top of the page and select Reports.
2. Click Review Blockers (under the 'Reports' sub-tab).
   - 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 a list of reviews for that 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.
2. Click JIRA Blockers (under the 'Reports' sub-tab).
3. Enter details for your JIRA server and project, and click Go.

The 'JIRA Blockers' report displays 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.

Screenshot: 'JIRA Blockers' 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, and which files, in your repository have been reviewed, 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 Stats panel
- Using the Changesets panel

Screenshot: The Review Coverage report
Opening the Review Coverage report

To open the Review Coverage report:

1. Click **Repositories** and choose a repository. The repository you chose sets the scope for the report.
2. If desired, navigate down the tree (in the lefthand panel) and click the desired path you want to view coverage on.
3. Click **Reports** in the secondary toolbar.
4. Click **Review Coverage** from the list of reports in the upper panel.

You can view coverage of any path by navigating 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 displays the following metrics for your selected repository:

- 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 Stats panel

The Individual Committer Stats panel 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

<table>
<thead>
<tr>
<th>Rank</th>
<th>Name</th>
<th>Reviewed</th>
<th>Lines</th>
</tr>
</thead>
<tbody>
<tr>
<td>1st</td>
<td>mwatson</td>
<td>100%</td>
<td>34/84</td>
</tr>
<tr>
<td>2nd</td>
<td>alyons</td>
<td>100%</td>
<td>67/67</td>
</tr>
<tr>
<td>3rd</td>
<td>amyers</td>
<td>100%</td>
<td>2/2</td>
</tr>
<tr>
<td>4th</td>
<td>sruiz</td>
<td>97%</td>
<td>1,140/1,178</td>
</tr>
<tr>
<td>5th</td>
<td>nvenegas</td>
<td>82%</td>
<td>256/326</td>
</tr>
<tr>
<td>6th</td>
<td>ahempel</td>
<td>61%</td>
<td>94/155</td>
</tr>
<tr>
<td>7th</td>
<td>csharkie</td>
<td>91%</td>
<td>654/731</td>
</tr>
<tr>
<td>8th</td>
<td>mquail</td>
<td>0%</td>
<td>0/154</td>
</tr>
<tr>
<td>9th</td>
<td>abutfeld</td>
<td>79%</td>
<td>659/833</td>
</tr>
<tr>
<td>10th</td>
<td>cmacneill</td>
<td>1%</td>
<td>2/187</td>
</tr>
<tr>
<td>11th</td>
<td>qcrain</td>
<td>58%</td>
<td>408/706</td>
</tr>
<tr>
<td>12th</td>
<td>tdavies</td>
<td>26%</td>
<td>306/1,182</td>
</tr>
</tbody>
</table>

All 66% 3,692/5,605
**Using the Changesets panel**

The Changesets 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 panel in the Review Coverage report**

![Changesets panel screenshot]

**Completing your review**

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

To complete your individual review, go to the review and click **Complete** at the top of the screen, next to the **Tools** menu:

![Complete button]

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

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

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

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

You have draft comments
Draft comments that aren't posted will be deleted.

View drafts
Delete drafts
Post drafts

Screenshot: Draft comments

Screenshot: Review complete

Review Complete

You have marked TEST-27 as completed.
The following reviewer is not yet finished:

☐ Geoff Crain

You can continue to add new comments and reply to comments until the moderator (Rosie Jameson) summarizes the review.

Reviews requiring your attention:

<table>
<thead>
<tr>
<th>Task</th>
<th>Review</th>
<th>Owner</th>
<th>Due</th>
</tr>
</thead>
<tbody>
<tr>
<td>Review</td>
<td>CR-FE-2170</td>
<td></td>
<td>03 Jul</td>
</tr>
<tr>
<td>Summarize</td>
<td>TEST-11</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
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 who are not logged in cannot send email, but they can view the text content of the review’s comments by clicking View Text that replaces Email Review in the Tools menu.
- 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. Choose Tools > Email Review (see Screenshot 1 below).
3. Specify the email recipients:

<table>
<thead>
<tr>
<th>To</th>
<th>You can enter multiple email addresses, separated by commas.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recipients</td>
<td>You can type usernames from your Crucible instance, to add them to the distribution list.</td>
</tr>
<tr>
<td>Send to Review Participants</td>
<td>Send the email to all reviewers.</td>
</tr>
</tbody>
</table>

4. Click Next.
5. You can modify both the Subject and Message parts of the email if required.
6. Click Send when you are ready. The ‘Status’ page confirms that your email has been sent.

Screenshot 1: The Email Review option in Crucible

Screenshot 2: The ‘Recipients’ Screen in Crucible
Email Review Comments

1. Specify recipients
2. Message details
3. Status

Recipients
Specify the people who should receive the comments of this review. You can specify a list of comma separated email addresses and/or specify one or more Crucible users.

To:

Recipients:
Start typing a user name then press enter to select.
- Send To Review Participants

Screenshot 3: The 'Message' Screen in Crucible

```
Email Review Comments

1. Specify recipients
2. Message details
3. Status

Message Details
This is the message as it will be sent. You can modify both the message content, as well as the subject of the email before sending it.

Subject: [TEST-27] test: All Comments

Message:
Subject: [TEST-27] Review: text
This is a list of all comments for TEST-27.
Review Summary: No summary
---
Title: text
Statement of Objectives:
documentation testing
State: Review
Author: Rosie Jamerson
Moderator: Rosie Jamerson
```

Screenshot 4: The Email Confirmation Screen in Crucible

Created by Atlassian in 2013. Licensed under a Creative Commons Attribution 2.5 Australia License.
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
Using RSS feeds in Crucible

Subscribing to an RSS feed

In Crucible, all pages with an activity stream, and any page that has a list of reviews, will have an RSS option. To access the RSS feed for a page, open the Tools menu at the top right corner of the screen, then choose 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.

<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>Biggest Text</td>
</tr>
<tr>
<td>h2.Bigger heading</td>
<td>Turns text into a heading at size 2.</td>
</tr>
<tr>
<td></td>
<td>Bigger heading</td>
</tr>
<tr>
<td>h3.Big heading</td>
<td>Turns text into a heading at size 3.</td>
</tr>
<tr>
<td></td>
<td>Big heading</td>
</tr>
<tr>
<td>h4.Normal heading</td>
<td>Turns text into a heading at size 4.</td>
</tr>
<tr>
<td></td>
<td>Normal heading</td>
</tr>
</tbody>
</table>
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 underlined.</td>
</tr>
<tr>
<td>??citation??</td>
<td>Makes text appear in —citation form.</td>
</tr>
<tr>
<td>-strikethrough-</td>
<td>Makes text appear struck through.</td>
</tr>
<tr>
<td>^superscript^</td>
<td>Makes text appear in superscript.</td>
</tr>
<tr>
<td><del>subscript</del></td>
<td>Makes text appear in subscript.</td>
</tr>
<tr>
<td>{{monospaced}}</td>
<td>Placing double curly-brackets around text makes it appear monospaced.</td>
</tr>
</tbody>
</table>

bq. Block Quote

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

Example:

```
Some block quoted text
```

{quote}

here is quoteable content to be quoted

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

Example:

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

{color:red}

look ma, red text!

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>
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>
</tbody>
</table>
| [Atlassian Crucible|http://atlassian.com] | 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. 

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). |
| [mailto:mail@example.com] | Creates a link to an email address. 
Example: mail@example.com |
| [file:///c:/temp/foo.txt] [file:///z:/file/on/network/share.txt] | Creates a download link to a file on your computer or on a network share that you have mapped to a drive. To access the file, you must right click on the link and choose "Save Target As". |
| {anchor:anchorname} | Creates a bookmark anchor inside the page. You can then create links directly to that anchor. So a link like this: [My Page#here] will link to wherever in "My Page" there is an {anchor:here} macro, and the link [#there] will link to wherever in the current page there is an {anchor:there} macro. |

Lists

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

<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 rows and columns, with a header row if required.

<table>
<thead>
<tr>
<th>Notation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</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>
</table>
| {noformat}                 | 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:  

This is a no-formatted piece of text, so *no* __formatting__ is done here. |
Embraces a block of text within a fully customizable panel. The optional parameters you can define are as follows.

- **title**: Title of the panel
- **borderStyle**: The style of the border this panel uses (solid, dashed and other valid CSS border styles)
- **borderColor**: The color of the border this panel uses
- **borderWidth**: The width of the border this panel uses
- **bgColor**: The background color of this panel
- **titleBGColor**: The background color of the title section of this panel

Examples:

```
Some text in a basic panel

My Title

Some text with a title
```

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:**

```
Bar.java

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

**A basic display with XML code:**

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

---

**Miscellaneous Markup Features**

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

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

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

To learn about these features, see the following pages:
- Using Progress Tracking
- Using Time Tracking

Using Progress Tracking

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

On this page:
- How progress tracking works in Crucible
- Viewing the progress tracking totals
- How to adjust progress tracking on a review
- Adjusting settings for progress tracking
- Further reading

How progress tracking works in Crucible

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

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

Viewing the progress tracking totals

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

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

Screenshot: Viewing the Progress Tracking Totals

How to adjust progress tracking on a review

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

Screenshot: Marking a File as Unread
Adjusting settings for progress tracking

Progress tracking is a configurable user preference – choose Profile settings from you user menu. In the Review settings sub-section. Auto-mark files as read is on by default. When Auto-mark files as read is set to off, you have to mark files as read or unread yourself.

Further reading

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

Using Time Tracking

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

On this page:

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

How time tracking works in Crucible

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

Screenshot: Crucible Time Tracking
How to adjust the time tracked on a review

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

Viewing the time tracking totals

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

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.

Further reading

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

**JIRA integration in Crucible**

**JIRA** is Atlassian's issue tracking and project management application.

When Crucible is integrated with JIRA, you and your team get all the benefits described on this page.

In Crucible, you can:

- Link your Crucible review to a JIRA issue
- Create a JIRA issue from a review comment
- Transition JIRA issues from within Crucible
• See issues from multiple instances of JIRA

In JIRA, you can:

• Create a Crucible review directly from an issue in JIRA

Related pages:

• Creating a review from JIRA
• Creating JIRA issues from the review
• Transitioning JIRA issues
• Linking Crucible to JIRA

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. Read more.

Link your Crucible review to a JIRA issue

When creating, or editing, your review, Crucible will suggest a JIRA issue that can be linked to the review, if a JIRA issue key is found in the review title. You can:

• click the suggested JIRA issue key, to link it to the review
• delete the suggested JIRA issue and specify a different issue key and click Link to save it.

Create a JIRA issue from a review comment

When viewing any review comment (general, file, inline), you can click Create Issue in the comment to create a JIRA issue. Crucible suggests the JIRA instance, project and issue type, but you can modify these. This requires JIRA 5.0, or later, and is disabled if Crucible is integrated with an earlier version of JIRA.

See Creating JIRA issues from the review for more details.
Transition JIRA issues from within Crucible

For Crucible reviews that have linked JIRA issues, you can advance the JIRA workflow for the issue from within Crucible. You can do this at any time by clicking the linked issue, or when you close the review:

See Transitioning JIRA issues for more details.

See issues from multiple instances of JIRA

Crucible can link to more than one JIRA server at a time, so different teams can work with their own projects in
Summarising and closing the review

Summarize is an optional step before closing a review.

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

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

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

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

Normally, we recommend that you wait for all reviewers to complete their reviews, before summarizing or closing the review.

The reviews that the reviewers have completed will be in your Ready to Close menu on the Dashboard.

To summarize a review,

- Click Summarize at the right of the screen.
- Optionally enter a summary of the review.
- If you have no further comments to add, click Close Review; otherwise, click Continue Without Closing.
- On clicking Summarize, you may be prompted to confirm the action if there are incomplete reviews or draft comments in the review. These are warnings, however; the review can still be summarized and closed.

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

Once the review is in the 'Summarize' state, the moderator can optionally add a review summary, for example, to describe the outcomes/tasks/etc:
The summary is sent to all participants and is displayed at the top of the closed review.

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

See:

- Using Review Reminders
- Moving a Review to Another Project
- Deleting a review

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.

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
Transitioning JIRA issues

When Crucible is linked to JIRA, you can advance the workflow for a JIRA issue directly from within Crucible.

You can transition a JIRA issue in two ways:

- Transitioning any JIRA issue at any time
- Transitioning a linked JIRA issue when closing the review

Transitioning any JIRA issue at any time

You can easily transition a JIRA issue at any time from within Crucible. Click on a JIRA issue link anywhere in Crucible to see a dialog with the available workflow steps:

Click on a step in the dialog, and complete any displayed fields as required. If there are custom required fields that are unsupported by Crucible, just click Edit this field in JIRA to transition the issue directly in JIRA.

Transitioning a linked JIRA issue when closing the review

When closing a Crucible review you may also want to close a JIRA issue that is linked to that review.

In the Review Summary screen, click Close near the top right. In the ‘Closed’ dialog, the available workflow transitions for the linked JIRA issue appear in the Transition issue dropdown:
Choose a step from the dropdown, and click **Close**.

**Notes**

- JIRA administrators can activate or deactivate JIRA issue transitions (the workflow) from the JIRA administration interface.
- Only the transitions accessible by the user are displayed.
- The list of available transitions only appears if the user has visibility to any available workflow transitions.

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

**Crucible releases**

**Crucible 3.1**

27 August 2013

- Dashboard improvements
- QuickNav and QuickSearch improvements
- New JIRA issue dialog
- Transition JIRA issues from within Crucible
- Small improvements: native SVN 1.7, OpenJDK
- More in the release notes and changelog

See the [Crucible upgrade guide](#).

**Crucible 3.0**

30 May 2013

- Redesigned user experience
- Review CLI tool
- Iterative pre-commit reviews
- Platform upgrades: Jetty 8, Infinity 3 DB
- Optimised indexing for new SVN repositories
- Small improvements
- More in the release notes and changelog

See the [Crucible upgrade guide](#).

**Crucible 2.10**

15 January 2013

- Inline issue creation
- Repository management REST API
- Repository review creation limit
- More in the release notes
- [Crucible 2.10 upgrade guide](#)
Crucible 2.9

14 November 2012

- Simpler JIRA integration
- Better performance of the Reviews tab in JIRA
- Faster review creation for large teams
- More in the release notes
- Crucible 2.9 upgrade guide

Crucible 2.8

15 August 2012

- Mentions
- Shares
- Improved performance for the projects listing
- Support for Subversion 1.7
- End of life announcements
- More in the release notes
- Crucible 2.8 Upgrade Guide

Crucible 2.7

7 September 2011

- JIRA Transitions in Crucible
- Review Reminders
- Small Improvements
- More in the release notes
- Crucible 2.7 Upgrade Guide

Crucible 2.6

6 June 2011

- New Quick Search
- HTML Emails for Reviews
- Dashboard and Navigation Improvements
- SQL Server Support
- Oracle Support
- Review Creation without Metadata Changes
- Improved Patch Anchoring
- More in the release notes
- Crucible 2.6 Upgrade Guide

Crucible 2.5

8 February 2011

- Oracle (Beta)
- Redesigned Activity Stream
- Improved Header
- Comment Notification Batching
- More in the release notes
- Crucible 2.5 Upgrade Guide

Crucible older release notes (click to expand)

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 the release notes
- Crucible 2.4 Upgrade Guide

Crucible 2.3

26 May 2010

- Snippet Reviews
- Changeset Discussions
- Mercurial SCM Alpha
- Review Coverage report
- Revamped Installation Process
- Gadgets
- More in the release notes
- Crucible 2.3 Upgrade Guide

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 the release notes
- Crucible 2.2 Upgrade Guide

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 the release notes
- Crucible 2.1 Upgrade Guide

Crucible 2.0

30 June 2009

- Support for iterative reviews
- New User Interface
- Indicators for read/unread comments
- Enhanced JIRA integration
- More in the release notes
- Crucible 2.0 Upgrade Guide

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 the release notes
- Crucible 1.6 Upgrade Guide

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 the release notes
- Crucible 1.5 Upgrade Guide

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 the release notes
- Crucible 1.2 Upgrade Guide

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 the release notes
- Crucible 1.1 Upgrade Guide

Crucible 1.0.x Changelog

_security advisories (click to expand)_

- FishEye and Crucible Security Advisory 2013-07-16
- 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

This page describes how to upgrade to a new version of Crucible.
Upgrade steps

This section provides general instructions for upgrading Crucible. See also the specific notes on this page for the version of Crucible you are upgrading to. We strongly recommend that you upgrade Crucible by following these steps:

- Back up your entire Crucible instance (see Backing up and restoring Crucible data):
  - 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.
- Read the Release Notes and Changelog and version-specific Upgrade Guides for the version you are upgrading to, as well as for any versions you are skipping.
- Read the Supported platforms page for the full list of supported platforms for Crucible.
- Download the Crucible zip file.

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

- Click here to expand...

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, using bin\stop.bat or bin\stop.sh from the <FishEye home directory>.
2. Make a backup of your FISHEYE_INST directory.
3. Download FishEye or Crucible.
4. Extract the new FishEye/Crucible version to a new directory.
5. Leave your FISHEYE_INST environment variable set to its existing location. Both FishEye and Crucible use this variable.
   - Please be aware that jar files in the FISHEYE_INST/lib directory may conflict with those required for FishEye's normal operation. Jar files in this directory should be limited to those which provide functionality not provided by FishEye (e.g. database drivers).
6. Start FishEye/Crucible from the new installation directory by running bin/run.sh. (Use run.bat on Windows.)
7. Follow any version-specific instructions found in the FishEye upgrade guide or Crucible upgrade guide.

Method 2: Without a FISHEYE_INST directory

Click here to expand...

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 server, using bin\stop.bat or bin\stop.sh from the FishEye home directory.
4. Copy <FishEye home directory>/config.xml to <New FishEye home directory>.
5. Delete the following directories from the <New FishEye home directory>/var directory:
   * <New FishEye home directory>/var/cache
   * <New FishEye home directory>/var/data
   * <New FishEye home directory>/var/log
6. Copy (or move) the following directories from <FishEye home directory>/var to <New FishEye home directory>/var:
   * <FishEye home directory>/var/cache
   * <FishEye home directory>/var/data
   * <FishEye home directory>/var/log
   DO NOT include the following directories when you do that:
   * <FishEye home directory>/var/osgi-cache
   * <FishEye home directory>/var/plugins
   * <FishEye home directory>/var/tmp
7. Delete the <New FishEye home directory>/cache directory.
8. Copy (or move) the <FishEye home directory>/cache directory to <New FishEye home directory>/cache.
9. Start FishEye/Crucible from the new installation by running <New FishEye home directory>/bin/run.sh (Use run.bat on Windows.)
10. Follow any version-specific instructions found in the FishEye upgrade guide or Crucible upgrade guide.

Method 3: Without a FISHEYE_INST directory, but would like to set one up

Click here to expand...

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 server, using bin\stop.bat or bin\stop.sh from the FishEye home directory.
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. If it exists, copy the <FishEye home directory>/data directory to the FISHEYE_INST directory.
8. Extract the new FishEye/Crucible archive into a directory such as <New FishEye home directory>.
9. 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.
10. Follow any version-specific instructions found in the FishEye upgrade guide or Crucible upgrade guide.

Upgrading to Crucible 3.1

Please also see the Upgrade steps section above, and read the End of Support Announcements for Crucible page.

Please note the following changes in Crucible 3.1:

**Crucible 3.1 Merge some per-repository Lucene indices into a global cross-repository Lucene index**

Crucible 3.1 has greatly improved performance and scalability for QuickSearch and QuickNav. To achieve this, the per-repository ‘METADATA’ Lucene indices will be moved into a single global cross-repository Lucene index. This means Crucible is able to search across more repositories in less time because now only a single search index needs to be queried instead of the previous N. Merging these indices into the single cross-repository index can be refreshed in two ways:

1. **Recommended**: As an upgrade task that is run automatically when Crucible 3.1 is run for the first time.
2. As an offline process on a separate staging server.

During the automatic upgrade task, Crucible is fully usable and functional, although search results for files, commits and committers may be incomplete.

In our testing we have found that the automatic upgrade task took 4 hours to complete on a Crucible instance with 144 repositories of different kinds, with 58 GB of data in the FISHEYE_INST folder (excluding logs). We are confident that the automatic upgrade task is suitable for the majority of production Crucible installations. *It is worth repeating that the instance was fully functional (reviews, JIRA Integration, Activity Streams, Charts etc) apart from Quick Nav and Quick Search during this time.*

Nevertheless, where required, we provide instructions for performing the reindex as an offline process on a separate staging server.

**Plugin Settings will be moved from the config.xml to the SQL database**

As of Crucible 3.1.0, plugin settings which were previously stored in the `<properties>` element inside config.xml will be stored in the SQL database. This includes settings for any bundled plugins such as ApplicationLinks, the UniversalPluginManager etc, and any 3rd party plugins.

An upgrade task is run on startup which will first insert all the properties found in config.xml into a new table in the SQL database. Once successful, the properties are removed from config.xml.

As part of this change, the RepositoryOptions.setProperties (Map<String, String>properties) and RepositoryOptions.getProperties() methods have been removed from our API. If you are using a plugin which uses either of these methods, you will need to update the plugin to a version which uses the Spring component PluginSettingsFactory. Plugins can use this to access the migrated global and per-repository properties that were previously available via the old RepositoryOptions API.

**Known issues for Crucible 3.1**

<table>
<thead>
<tr>
<th>Type</th>
<th>Key</th>
<th>Summary</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>🚹</td>
<td>CRUC-6567</td>
<td>Perforce native SCM management get branches completely wrong</td>
<td>➔ Open</td>
</tr>
<tr>
<td>🚹</td>
<td>CRUC-6521</td>
<td>File creation is not displayed as change</td>
<td>➔ Open</td>
</tr>
<tr>
<td>Issue</td>
<td>Description</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>-------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CRUC-6596</td>
<td>Resolving long content hashes from short content hashes in git patches goes over the whole contentHashDB for each hash in patch</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CRUC-6553</td>
<td>Upgrade to 3.0.x may fail if there are unprocessed cru_feindex_msg rows</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CRUC-6550</td>
<td>Code Review UI real estate misallocation</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CRUC-6528</td>
<td>User still suggested for review after disabling for him Crucible</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CRUC-6520</td>
<td>If the P4DIFF environment variable is set, the crucible.py review tool might not be able to get the patch properly</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CRUC-6487</td>
<td>ConstraintViolationException: could not insert: [com.cenqua.crucible.model.CommentReadStatus]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CRUC-6470</td>
<td>Inline issue creation: when not logged in to Crucible the review comment icon is not fetched</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CRUC-6447</td>
<td>Providing a FishEye licence does not allow repositories to be started until FishEye itself has been stopped and started</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Authenticate to retrieve your issues

Upgrading to Crucible 3.0

Please also see the Upgrade steps section above.

Please note the following changes in Crucible 3.0:

**Jetty 8**

Crucible 3.0 now uses Jetty 8 as its web server and Java servlet container. This change should be completely transparent when you upgrade to Crucible 3.0. However, if you have customised either your jetty-web.xml file, or the maxFormContentSize system property, you will need to update those in the new version. See Enabling Access Logging in FishEye and this FishEye KB page for more information.
Crucible 3.0 now uses the InfinityDB 3.0 database internally to provide improved performance for concurrent access to Crucible. This change is transparent to users in all respects.

**Pipelined indexing**

Crucible 3.0 introduces a new indexing approach that splits the repository indexing process into separate tasks that can be performed in a phased and concurrent way. Users will benefit from the way in which Crucible functionality, such as review creation, now becomes available as indexing progresses. This change is transparent to users in all other respects. See Pipelined indexing.

**Improved handling of user preferences with session cookies**

Upgrading may result in some users losing their preferences.

**SQL Server transaction isolation configuration**

We recommend a configuration change for SQL Server to use snapshot mode for the transaction isolation level – see Migrating to SQL Server. This change avoids occasional database deadlocks, and prevents performance warning messages in the logs and admin screens.

**Known issues for Crucible 3.0**

<table>
<thead>
<tr>
<th>Type</th>
<th>Key</th>
<th>Summary</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>CRUC-6567</td>
<td>Perforce native SCM management get branches completely wrong</td>
<td>Open</td>
</tr>
<tr>
<td></td>
<td>CRUC-6521</td>
<td>File creation is not displayed as change</td>
<td>Open</td>
</tr>
<tr>
<td></td>
<td>CRUC-6596</td>
<td>Resolving long content hashes from short content hashes in git patches goes over the whole contentHashDB for each hash in patch</td>
<td>Open</td>
</tr>
<tr>
<td></td>
<td>CRUC-6553</td>
<td>Upgrade to 3.0.x may fail if there are unprocessed cru_feindex_msg rows</td>
<td>Open</td>
</tr>
<tr>
<td></td>
<td>CRUC-6550</td>
<td>Code Review UI real estate misallocation</td>
<td>Open</td>
</tr>
<tr>
<td></td>
<td>CRUC-6528</td>
<td>User still suggested for review after disabling for him Crucible</td>
<td>Open</td>
</tr>
<tr>
<td></td>
<td>CRUC-6520</td>
<td>If the P4DIFF environment variable is set, the crucible.py review tool might not be able to get the patch properly</td>
<td>Open</td>
</tr>
<tr>
<td></td>
<td>CRUC-6487</td>
<td>ConstraintViolationException: could not insert: [com.cenqua.crucible.model.CommentReadStatus]</td>
<td>Open</td>
</tr>
</tbody>
</table>
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 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.**

**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.

**FISHEYE_INST**— this is an environment variable that defines the path to the FishEye data directory, which is used for installations where the FishEye instance data is stored in a separate location from the install (home) directory. Both Crucible and FishEye use this environment variable.

Read more about the **FISHEYE_INST** environment variable.

On this page:

- **Upgrade procedure**
  - Method 1: Without a FISHEYE_INST directory (default)
  - Method 2: Using a FISHEYE_INST directory
  - Method 3: No FISHEYE_INST directory, but wish to use one
  - Initial Crucible configuration

**Upgrade procedure**

- **We strongly recommend that you make a backup of your data before following one of the methods below.** Refer to making a backup.
- **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>, separately from your FishEye data directory, which has 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.

Method 1: Without a FISHEYE_INST directory (default)

1. Download Crucible.
2. Extract the new Crucible archive into a directory, which we will refer to as <Crucible directory>.
3. Shut down your existing FishEye server, if it is running.
4. Copy <FishEye home directory>/config.xml to <Crucible directory>.
5. Delete the <Crucible directory>/var directory.
6. Copy the <FishEye home directory>/var directory to <Crucible directory>/var.
7. Follow any version-specific instructions found in the Release Notes.
8. Start Crucible from the new installation by running <Crucible 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. Shut down your existing FishEye server, if it is running.
2. Make a backup of your FISHEYE_INST directory.
3. Download Crucible and unzip the archive into a directory, which we will refer to as <Crucible 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 <Crucible directory>/bin/run.sh. (Use run.bat on Windows).
6. Follow the initial configuration steps outlined below.

Method 3: No FISHEYE_INST directory, but wish to use one

1. Shut down your existing FishEye server, if it is running.
2. Create a new directory on your filesystem, then set up a FISHEYE_INST environment variable to point to the new directory.
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 <Crucible directory>.
7. Start Crucible from the new installation by running <Crucible directory>/bin/run.sh. (Use run.bat on Windows).
8. Follow the initial configuration steps outlined below.
9. 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 <Crucible directory> and FISHEYE_INST. Check that FISHEYE_INST is pointing to the correct directory - it should point to the location to where you copied the existing FishEye files in steps 3 and 4 above.

Initial Crucible configuration

1. You can access FishEye immediately by going to http://HOSTNAME:8060/ in a browser.
2. The first time you run FishEye after upgrading, you must enter your Crucible license key. To do this, update your Crucible license by going to the admin area and clicking System Info (under 'System Settings'). Click Edit License (under 'License') to 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/.
   • Click Users/Security under 'Global Settings' in the admin area.
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 3.1 release notes**

27 August 2013

Today we’re excited to release Crucible 3.1, which brings you a cleaner and faster interface. We’ve also added some further JIRA integration points.

If you are upgrading from an earlier version of Crucible, please read the Crucible upgrade guide. Please also read the End of Support Announcements for Crucible.

The Crucible 3.1 changelog is at the bottom of this page.

The new Dashboard

Crucible's new dashboard makes your pending reviews the highest priority, putting them right at the top and clearly indicating their urgency. The repository and project navigator lets you jump to your recent content (try pressing Enter as soon as you launch the dashboard), and includes filters and keyboard shortcuts that facilitate navigation, especially when your organisation has hundreds of repositories.

**QuickNav and QuickSearch improvements**

Previously in Crucible, QuickSearch was limited to searching 100 repositories concurrently. Now, in Crucible 3.1, you can search across files and directories, commit messages, and committers, for every one of your repositories. And it’s 40% faster. And you can order search results by date. Furthermore, QuickNav now uses the same indexes as QuickSearch, so it’s faster too, and it produces the same results as QuickSearch:

Note that upgrading to Crucible 3.1 requires an upgrade to the cross-repository index. See the Crucible upgrade guide for details.

New JIRA issue dialog
Now, when you click on any linked JIRA issue key in Crucible, you’ll see a dialog that displays a summary of the issue details. From that dialog, you can transition the workflow for the issue, and easily click through to edit the issue in JIRA:

Transition JIRA issues from within Crucible

Previously in Crucible, you could transition a JIRA issue linked to a review at the time that you closed the review. Now you can advance the workflow for any JIRA issue mentioned in Crucible at any time, right from within Crucible – just click the issue link to see the transitions available to you:

Read more about JIRA issue transitions in Crucible ...

Small improvements

Simplified reminder for overdue reviewers

Alerting reviewers about an overdue review is now a simple 1-click action:

Improved browsing performance

The display of the file source page is now 60% faster in Crucible 3.1.

Native SVN 1.7 support

Crucible 3.1 now supports the native JavaHL 1.7 SVN client. It does not support the native JavaHL 1.6 SVN client. See Native support for SVN 1.7 for more details.

OpenJDK support

Crucible 3.1 now supports OpenJDK 1.7. See Supported platforms for details.
Change log

This section will contain information about the Crucible 3.1 minor releases as they become available. These releases will be free to all customers with active Crucible software maintenance.

If you are upgrading from an earlier version of Crucible, please read the Crucible upgrade guide.

The issues listed below are just the highlights of all those that have been resolved for the Crucible 3.1.x releases.

17 September 2013 - Crucible 3.1.3

Fixes a bug which can cause review indexing to fail when upgrading from versions prior to 3.0.

<table>
<thead>
<tr>
<th>Type</th>
<th>Key</th>
<th>Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>CRUC-6595</td>
<td>Some Ajax calls never return, eventually timeout with a 502 proxy error</td>
</tr>
</tbody>
</table>

 Authenticate to retrieve your issues

11 September 2013 - Crucible 3.1.2

- Few XSS security issues addressed

29 August 2013 - Crucible 3.1.1

<table>
<thead>
<tr>
<th>Type</th>
<th>Key</th>
<th>Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>CRUC-6584</td>
<td>Unable to create reviews after upgrade to 3.1</td>
</tr>
</tbody>
</table>

 Authenticate to retrieve your issues

27 August 2013 - Crucible 3.1.0

<table>
<thead>
<tr>
<th>Type</th>
<th>Key</th>
<th>Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>+</td>
<td>CRUC-6384</td>
<td>Allow issue transitions for issues raised in Inline Issue Creation</td>
</tr>
<tr>
<td></td>
<td>CRUC-6530</td>
<td>Remove Crucible Transition error as a warning</td>
</tr>
<tr>
<td></td>
<td>CRUC-6534</td>
<td>Type when starting FishEye/Crucible</td>
</tr>
<tr>
<td></td>
<td>CRUC-6525</td>
<td>Upgrade to 3.x release occasionally fails when deduplicating paths in db</td>
</tr>
<tr>
<td></td>
<td>CRUC-6417</td>
<td>Better handling of Authentication Errors when manually linking a Review to a JIRA Issue</td>
</tr>
<tr>
<td></td>
<td>CRUC-6416</td>
<td>Optimize number of database queries for Crucible instances with Many Projects and Permission Schemes</td>
</tr>
</tbody>
</table>
Crucible 3.0 release notes

30 May 2013

Today we’re excited to release Crucible 3.0, which introduces huge indexing performance gains – especially for Subversion repositories – and significant upgrades to how you perform pre-commit reviews. We think development teams will be able to work faster, every day.

If you are upgrading from an earlier version of Crucible, please read the Crucible upgrade guide.

The Crucible 3.0 changelog is at the bottom of this page.

Try it for FREE

Redesigned user experience

We’ve made a few key design changes to help you work faster with your reviews:

New global header

The new header across the top of the page makes it quicker to access recent repositories, projects, people and reviews, and has the large Create review button at top center, right where you can find it.

Application navigator

The new application navigator, at the left of the header, connects you directly to your other applications, such as JIRA and Bamboo. Now you can switch between Crucible and JIRA – or any other Atlassian application – all from the Crucible header. No more bookmarks in your browser; we do the job for you. Admins can easily configure which apps appear in the navigator – just click Application navigator in the admin area.

JIRA Reviews tab redesign

We wanted to simplify the experience but still give you the important information you need to make decisions around your issues. We think you’ll like it:
User documentation for Crucible 3.1

Activity

<table>
<thead>
<tr>
<th>All</th>
<th>Comments</th>
<th>Work Log</th>
<th>History</th>
<th>Activity</th>
<th>Commits</th>
<th>Source</th>
<th>Reviews</th>
<th>Builds</th>
</tr>
</thead>
</table>

CR-IE-7319 CLOSED

[View issue](https://jira.atlassian.com/browse/CR-IE-7319)

Review CLI tool

Crucible now ships with a command line tool that allows you to quickly create reviews in Crucible directly from your terminal. The tool takes the uncommitted changes in your workspace and creates a review for them in Crucible. You can use the tool to create new reviews, and to update existing pre-commit reviews with new patches. [Read more.]

Iterative pre-commit reviews

Crucible now supports iterative reviews for both post-commit and pre-commit reviews. This allows you to update the review with new versions of files, and changesets (for post-commit reviews) or patches (for pre-commit reviews) that have been created after the review was started.

Iterative reviews allow your team to discuss changing code in the context of a single review. This lets the reviewers see all the related changes together, and to more easily keep track of comments and defects. [Read more.]

Create reviews sooner for new SVN repositories

We've introduced a new approach that splits the indexing process into separate tasks that can be performed in a phased and concurrent way. This allows you to start using core functionality in Crucible, such as creating reviews, up to 15X sooner than in Crucible 2.10. You can get on with your work, while Crucible quietly completes the fine details of indexing in the background. [Read more.]

Small improvements

Faster browsing for large teams

Crucible now uses the InfinityDB 3.0 internal database to provide faster concurrent access. Your team gets a better browsing experience!

Quicker review creation

We've optimised some SQL statements in Crucible. You spend less time creating, or modifying, reviews.

Better review creation from JIRA when Stash is integrated

We've improved the integration between JIRA and Crucible. Now you can create a review for all the commits related to a JIRA issue, even when Atlassian Stash is integrated with JIRA.

Created by Atlassian in 2013. Licensed under a Creative Commons Attribution 2.5 Australia License.
Diff hunk shortcuts

We've changed the 'j' and 'k' keyboard shortcuts to now work with diff hunks when looking at a review, so as to be consistent with their new behaviour in FishEye.

Improved handling of user preferences with session cookies

We've made session cookies more robust, but this may result in your existing preferences being lost at upgrade time.

REST API change

The 'groups-v1' REST endpoint now requires the user to be an administrator – it will fail authentication and return a 403 otherwise.  Read more...

Change log

This section will contain information about the Crucible 3.0 minor releases as they become available. These releases will be free to all customers with active Crucible software maintenance.

If you are upgrading from an earlier version of Crucible, please read the Crucible upgrade guide.

The issues listed below are just the highlights of all those that have been resolved for the Crucible 3.0.x releases.

23 July 2013 - Crucible 3.0.3

<table>
<thead>
<tr>
<th>Type</th>
<th>Key</th>
<th>Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>CRUC-6556</td>
<td>Error editing permission scheme action containing deleted user</td>
</tr>
<tr>
<td></td>
<td>CRUC-6552</td>
<td>Searching over REST for a JIRA Project key doesn't look at Review Linked Issues</td>
</tr>
<tr>
<td></td>
<td>CRUC-6537</td>
<td>Deleting and recreating projects with keys which differ only in case can cause NonUniqueResultException on Viewing Review</td>
</tr>
</tbody>
</table>

2 July 2013 - Crucible 3.0.2

<table>
<thead>
<tr>
<th>Type</th>
<th>Key</th>
<th>Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key</td>
<td>Summary</td>
<td></td>
</tr>
<tr>
<td>----------</td>
<td>-------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>CRUC-6503</td>
<td>Crucible upgrade may fail in the middle if DB socket timeout too short</td>
<td></td>
</tr>
<tr>
<td>CRUC-6555</td>
<td>Can't use distinct in queries on tables with CLOB types in Oracle</td>
<td></td>
</tr>
<tr>
<td>CRUC-6504</td>
<td>Review pre-commit tool can fail to include all the lines for an added file with Perforce</td>
<td></td>
</tr>
<tr>
<td>CRUC-6416</td>
<td>Optimize number of database queries for Crucible instances with Many Projects and Permission Schemes</td>
<td></td>
</tr>
</tbody>
</table>

**Authenticate** to retrieve your issues

### 6 June 2013 - Crucible 3.0.1

<table>
<thead>
<tr>
<th>Key</th>
<th>Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>CRUC-6491</td>
<td>duplicated null cru_stored_path rows leads to 3.0 upgrade problems</td>
</tr>
</tbody>
</table>

**Authenticate** to retrieve your issues

### 28 May 2013 - Crucible 3.0.0

<table>
<thead>
<tr>
<th>Key</th>
<th>Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>CRUC-6434</td>
<td>Add disclaimer that &quot;Create crucible review&quot; is only available with FishEye integration</td>
</tr>
<tr>
<td>CRUC-6414</td>
<td>Update on 'Editing a Project' documentation</td>
</tr>
<tr>
<td>CRUC-6271</td>
<td>Update on Using Gadgets in Crucible documentation</td>
</tr>
<tr>
<td>CRUC-6410</td>
<td>Allow /filter/details to include a date range filter</td>
</tr>
<tr>
<td>CRUC-6361</td>
<td>Change semantics of j/k shortcuts in Crucible, be consistent with the new diff navigation in Fisheye</td>
</tr>
<tr>
<td>CRUC-6394</td>
<td>Crucible Project Repository Link is Broken if Leading Slash Is Omitted</td>
</tr>
<tr>
<td>CRUC-6391</td>
<td>Documentation Update for JIRA Integration</td>
</tr>
<tr>
<td>CRUC-6349</td>
<td>Crucible and FishEye Page Contains Outdated Information</td>
</tr>
</tbody>
</table>
CRUC-4518  Crucible regularly hangs and is very slow
CRUC-2998  Iterative pre-commit reviews
CRUC-1452  REST API should notify client if session token is invalid
CRUC-6466  Dismiss barracuda’s feature discovery message when it is closed.
CRUC-6463  Multiple open transactions in idle instance
CRUC-6454  Scheduled user synchronization doesn't close DB sessions
CRUC-6412  Stream has already been closed on getting Notifications
CRUC-6408  URLs in comments with # followed by ! and other characters are not correctly generated.
CRUC-6398  Unable to link to JIRA issues containing a project key longer than 10 characters, containing underscores _
CRUC-6395  Inconsistent Repository Links on Project Page
CRUC-6390  Documentation Update for Smart Commits
CRUC-6385  An SQLException can corrupt a connection in the connection pool and cause Crucible to became unstable

Authenticate to retrieve your issues

9 more issues

Crucible 2.10 release notes

15 January 2013

With great pleasure, Atlassian presents Crucible 2.10, which introduces the best JIRA integration we have ever built.

- See the change log for Crucible 2.10.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 2.10 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.
- JIRA 5.0 integration. The features described below are supported by JIRA 5.0, or later, with the latest
Highlights of this release:

- Inline Issue Creation
- Repository Management REST API
- Repository review creation limit
- Other announcements

Providing feedback:

Please log your votes and issues. They help us decide what needs doing, and are much appreciated!

1

Inline Issue Creation

With Crucible 2.10 you can now create any type of issue from your reviews. You only need to link your Crucible and JIRA instances to get it working. Once this is done you can create issues in any JIRA project you have access to.

2

Repository Management REST API

Crucible 2.10 lets you optimize your instance by using the REST API to trigger indexing when it's necessary. Instead of configuring Crucible to poll each repository every minute, you can add a post-commit or post-receive hook to your repositories which will ping Crucible when new commits need to be indexed.

If your repositories are hosted on Bitbucket or Github you will find corresponding Service Hooks in the settings of your repositories which will do the job for you.
3

Repository review creation limit

Crucible 2.10 now has a limit to prevent oversized reviews from being created during wrong manipulations. We have set the limit fairly high at 800 file revisions, a number way above the recommended number of files per review. The limit will help to prevent your instance from having performance issues.

4

Other announcements

- Improved performance of the repositories listing in the administration interface.
- Snippets share the same permission as reviews.
- You can now easily step through the diffs in reviews via shortcuts.

The Crucible 2.10 team

Development

Core team
Brendan Humphreys
Conor MacNeill
Geoff Crain
Lukasz Pater
Maciej Swinarski
Piotr wity wiciki
Richard Stephens
Tom Davies
Valery Trubnikov

Team lead
Crucible 2.10 changelog

This page contains information about the Crucible 2.10 minor releases. FishEye license holders should also check the FishEye 2.10 Changelog.
See the Crucible 2.10 release notes for details of what’s new in 2.10.

Please read the Crucible 2.10 Upgrade Guide before upgrading to any of the minor releases below.

17 July 2013 - Crucible 2.10.7

<table>
<thead>
<tr>
<th>Type</th>
<th>Key</th>
<th>Summary</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>🔄</td>
<td>CRUC-6555</td>
<td>Can’t use distinct in queries on tables with CLOB types in Oracle</td>
<td>🔄 Closed</td>
</tr>
</tbody>
</table>

Authenticate to retrieve your issues

05 July 2013 - Crucible 2.10.6

Created by Atlassian in 2013. Licensed under a Creative Commons Attribution 2.5 Australia License.
<table>
<thead>
<tr>
<th>Type</th>
<th>Key</th>
<th>Summary</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>🧑‍💼</td>
<td>CRUC-6416</td>
<td>Optimize number of database queries for Crucible instances with Many Projects and Permission Schemes</td>
<td>🧑‍💼 Closed</td>
</tr>
<tr>
<td>🧑‍💼</td>
<td>CRUC-6463</td>
<td>Multiple open transactions in idle instance</td>
<td>🧑‍💼 Closed</td>
</tr>
<tr>
<td>🧑‍💼</td>
<td>CRUC-6394</td>
<td>Crucible Project Repository Link is Broken if Leading Slash is Omitted</td>
<td>🧑‍💼 Closed</td>
</tr>
<tr>
<td>🧑‍💼</td>
<td>CRUC-6412</td>
<td>Stream has already been closed on getting Notifications</td>
<td>🧑‍💼 Closed</td>
</tr>
<tr>
<td>🧑‍💼</td>
<td>CRUC-6408</td>
<td>URLs in comments with # followed by ! and other characters are not correctly generated.</td>
<td>🧑‍💼 Closed</td>
</tr>
<tr>
<td>🧑‍💼</td>
<td>CRUC-6395</td>
<td>Inconsistent Repository Links on Project Page</td>
<td>🧑‍💼 Closed</td>
</tr>
<tr>
<td>🧑‍💼</td>
<td>CRUC-5238</td>
<td>URLs with # in comments do not render properly</td>
<td>🧑‍💼 Closed</td>
</tr>
<tr>
<td>🧑‍💼</td>
<td>CRUC-2747</td>
<td>Review details says &quot;Reviewer is 100% complete&quot; when she hasn't completed the review</td>
<td>🧑‍💼 Closed</td>
</tr>
</tbody>
</table>

**06 Jun 2013 - Crucible 2.10.5**

<table>
<thead>
<tr>
<th>Type</th>
<th>Key</th>
<th>Summary</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>🧑‍💼</td>
<td>CRUC-6416</td>
<td>Optimize number of database queries for Crucible instances with Many Projects and Permission Schemes</td>
<td>🧑‍💼 Closed</td>
</tr>
</tbody>
</table>

**02 Apr 2013 - Crucible 2.10.4**

<table>
<thead>
<tr>
<th>Type</th>
<th>Key</th>
<th>Summary</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>🧑‍💼</td>
<td>CRUC-6394</td>
<td>Crucible Project Repository Link is Broken if Leading Slash is Omitted</td>
<td>🧑‍💼 Closed</td>
</tr>
<tr>
<td>🧑‍💼</td>
<td>CRUC-6412</td>
<td>Stream has already been closed on getting Notifications</td>
<td>🧑‍💼 Closed</td>
</tr>
<tr>
<td>🧑‍💼</td>
<td>CRUC-6408</td>
<td>URLs in comments with # followed by ! and other characters are not correctly generated.</td>
<td>🧑‍💼 Closed</td>
</tr>
<tr>
<td>🧑‍💼</td>
<td>CRUC-6395</td>
<td>Inconsistent Repository Links on Project Page</td>
<td>🧑‍💼 Closed</td>
</tr>
<tr>
<td>🧑‍💼</td>
<td>CRUC-5238</td>
<td>URLs with # in comments do not render properly</td>
<td>🧑‍💼 Closed</td>
</tr>
<tr>
<td>🧑‍💼</td>
<td>CRUC-2747</td>
<td>Review details says &quot;Reviewer is 100% complete&quot; when she hasn't completed the review</td>
<td>🧑‍💼 Closed</td>
</tr>
</tbody>
</table>

**19 Mar 2013 - Crucible 2.10.3**

<table>
<thead>
<tr>
<th>Type</th>
<th>Key</th>
<th>Summary</th>
<th>Status</th>
</tr>
</thead>
</table>
Unable to link to JIRA issues containing a project key longer than 10 characters, containing underscores

Authenticated to retrieve your issues

25 Feb 2013 - Crucible 2.10.2

<table>
<thead>
<tr>
<th>Type</th>
<th>Key</th>
<th>Summary</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>CRUC-6385</td>
<td>An SQLException can corrupt a connection in the connection pool and cause Crucible to become unstable</td>
<td>Closed</td>
<td></td>
</tr>
<tr>
<td>CRUC-6383</td>
<td>Inline Issue Creation doesn't remember the last issue type selected</td>
<td>Closed</td>
<td></td>
</tr>
<tr>
<td>CRUC-6380</td>
<td>Reinstate the Star/Favourite link for comments</td>
<td>Closed</td>
<td></td>
</tr>
<tr>
<td>CRUC-6350</td>
<td>JIRA errors being displayed inline on review comments</td>
<td>Closed</td>
<td></td>
</tr>
<tr>
<td>CRUC-6345</td>
<td>Can not remove reviewers in IE</td>
<td>Closed</td>
<td></td>
</tr>
<tr>
<td>CRUC-6268</td>
<td>When creating issue from comment summary put by the user gets lost on JIRA/project change</td>
<td>Closed</td>
<td></td>
</tr>
<tr>
<td>CRUC-6246</td>
<td>User hovers are broken in IE9 (and probably 8 and 10 as well)</td>
<td>Closed</td>
<td></td>
</tr>
</tbody>
</table>

Authenticated to retrieve your issues

29 Jan 2013 - Crucible 2.10.1

<table>
<thead>
<tr>
<th>Type</th>
<th>Key</th>
<th>Summary</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>CRUC-6274</td>
<td>Update Add Comments related Crucible document</td>
<td>Closed</td>
<td></td>
</tr>
<tr>
<td>CRUC-6344</td>
<td>Change Inline Issue Creation upgrade task warning</td>
<td>Closed</td>
<td></td>
</tr>
<tr>
<td>CRUC-6342</td>
<td>LazyInitializationException during indexing of repositories containing</td>
<td>Closed</td>
<td></td>
</tr>
</tbody>
</table>

Created by Atlassian in 2013. Licensed under a Creative Commons Attribution 2.5 Australia License.
### Changeset Comments

**CRUC-6333**

IIC Plugin makes unneeded AJAX requests on pages other than the review page.

- **Closed**

---

<table>
<thead>
<tr>
<th>Type</th>
<th>Key</th>
<th>Summary</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>$</td>
<td>CRUC-6282</td>
<td>If an Entity Link is present between a JIRA Project and a Crucible Project, then a review created from the JIRA Source Tab should default to the entity linked Crucible Project.</td>
<td>Closed</td>
</tr>
<tr>
<td></td>
<td>CRUC-6351</td>
<td>Managing Add-Ons Page Refers to Add-On That Is Not Compatible With Latest Versions</td>
<td>Closed</td>
</tr>
<tr>
<td></td>
<td>CRUC-6143</td>
<td>When Creating Jira Issues From a Code Review Provide Capability to Handle REQUIRED Fields</td>
<td>Closed</td>
</tr>
<tr>
<td></td>
<td>CRUC-6031</td>
<td>Need keyboard shortcuts for moving to previous, current, and next diff.</td>
<td>Closed</td>
</tr>
<tr>
<td></td>
<td>CRUC-5847</td>
<td>The Graph should display % and also value.</td>
<td>Closed</td>
</tr>
<tr>
<td></td>
<td>CRUC-5628</td>
<td>Add Ability to Limit Number and/or Size of Crucible Reviews</td>
<td>Closed</td>
</tr>
<tr>
<td></td>
<td>CRUC-6267</td>
<td>No success/error notification when creating JIRA issue directly from comment</td>
<td>Closed</td>
</tr>
<tr>
<td></td>
<td>CRUC-6266</td>
<td>Smart commit plugin does not handle multi-line commit messages</td>
<td>Closed</td>
</tr>
<tr>
<td></td>
<td>CRUC-6259</td>
<td>Share review user suggestion contain deleted user</td>
<td>Closed</td>
</tr>
<tr>
<td></td>
<td>CRUC-6249</td>
<td>There is no pre-load handling for next diff?</td>
<td>Closed</td>
</tr>
</tbody>
</table>
User documentation for Crucible 3.1

CRUC-6248  Double infinite spinner, what does it mean?  ✔️ Closed
CRUC-6210  Deleted users are appearing in reviewer, moderator dropdown  ✔️ Closed
CRUC-6142  "Edit Review Details" end sometimes in "The server is not responding" but the server is ok  ✔️ Closed
CRUC-6072  FeCru REST API docs don't provide description of the XML tags.  ✔️ Closed
CRUC-6038  Error creating subtask in JIRA: com.atlassian.jira.rpc.exception.RemotePermissionException: This issue does not exist or you don't have permission to view it.  ✔️ Closed
CRUC-5716  GWT error when navigating away from loading page  ✔️ Closed
CRUC-5666  Snippet review link is displayed even if user doesn't have permission to create a review in any project  ✔️ Closed
CRUC-2667  Not able to create the subtask in JIRA if some of the fields are required in the subtask creation screen  ✔️ Closed
CRUC-2202  NPE thrown when trying to change project key  ✔️ Closed

Authenticate to retrieve your issues

Crucible 2.10 upgrade guide

Please refer to the FishEye 2.10 upgrade guide for important notes on upgrading to Crucible 2.10.

For details of the new features and improvements in this release, please read the Crucible 2.10 Release Notes.

Upgrading to Crucible 2.10

Crucible Subtask Creation has been deprecated

If you are currently using issue subtask creation please read the following section.

We have improved JIRA issue creation from Crucible with a feature called Inline Issue Creation. It uses JIRA
APIs introduced in JIRA 5.0, so it is not compatible with earlier versions. It also has some changed behaviour from the old "Crucible Subtask Creation".

**If you are using JIRA 4.4 or earlier**

The new feature will not be compatible with your JIRA version. You can revert back to the old behaviour if you want to continue being able to create Crucible subtasks from review comments.

Disabling of the plugin should be automatic on upgrade. Crucible will poll your linked JIRA servers to check their version. If a version earlier than JIRA 5.0 is detected, the 'Inline Issue Creation' plugin will be disabled.

If the plugin is not disabled you can revert to the old behaviour by clicking Manage Add-ons, under 'System Settings' in the Admin area. Click Show system plugins, find the 'Inline Issue Creation' plugin, and disable it. When you upgrade your JIRA server, you can come back here to re-enable the plugin.

**If you are using JIRA 5.0 or later**

Removed functionality:

- The ability to "Resolve" after the issue has been created has been removed. Users will now have to transition issues via the JIRA interface.
- Existing subtasks that have been created will still be visible, however you will not be able to resolve them.

New functionality:

- We now support various required fields, both built-in and custom field types.
- There is no longer any configuration required, except for the Application Link to JIRA.
- We now support 'normal' (non-subtask) issue creation, users can choose what Project and Issue Type to create in, and we support various required fields.
- There is now a report at the top of the review for all issues created.
- There is now a Remote Issue Link from the JIRA issue back to the comment.
  - Note that Issue Linking can be disabled in JIRA.
  - The user creating the issue must have the permission to create Remote Issue Links.
  - The latest version of the JIRA FishEye Plugin will hide these links from users without the View Source permission.

**Crucible 2.9 release notes**

**14 November 2012**

With great pleasure, Atlassian presents Crucible 2.9, which introduces the best JIRA integration we have ever built.

- See the change log for Crucible 2.9.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 2.9 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.
- JIRA 5.0 integration. The features described below are supported by JIRA 5.0, or later, with the latest version of the JiRA FishEye plugin.

Highlights of this release:

- Simpler JIRA integration
- Better performance of the Reviews tab in JIRA
- Faster review creation for large teams
- Other announcements
Providing feedback:

Please log your votes and issues. They help us decide what needs doing, and are much appreciated!

1

Simpler JIRA integration

With Crucible 2.9, linking your instance to JIRA is as simple as Plug & Play. As soon as you create an Application Link between your JIRA server and your Crucible instance, you get all the power of the JIRA / Crucible integration:

- See in JIRA the list of reviews corresponding to a specific issue
- Navigate to the related review from your issue page
- Get the JIRA data corresponding to issues mentioned in your review comments, titles and objectives

You no longer need to create and maintain multiple links between projects in JIRA and projects in Crucible. Maintaining your JIRA / Crucible integration has never been simpler! More...

Activity

<table>
<thead>
<tr>
<th>CR-FE-6832</th>
<th>Closed</th>
</tr>
</thead>
<tbody>
<tr>
<td>FECRU-2765</td>
<td>re-enable highlights for ie</td>
</tr>
<tr>
<td><a href="e32efafa766e915667497b33a10303821747274139">commit</a> [rev=FE-hg]: FECRU-2765</td>
<td>re-enable highlights for ie</td>
</tr>
</tbody>
</table>

Comment

2

Better performance of the Reviews tab in JIRA

When we changed the implementation of the Crucible / JIRA integration we also improved the performance of the Reviews Tab display in JIRA. You will see your reviews in JIRA faster, even without having to create any links between your JIRA project and your Crucible repositories. In other words, with Crucible 2.9 you spend less time administering your instance, while your users get all the relevant content in JIRA in a better way than before.
Faster review creation for large teams

Crucible is a tool that is built to help organizations and distributed teams perform code reviews. When you have large teams you can find that you always have a large number of reviews waiting to be closed. Prior to Crucible 2.9, this could slow down the review creation process because Crucible looked for other reviews where changesets might have been included.

With Crucible 2.9 we have removed the Suggested Reviews step during review creation. Not only does this accelerate the process, but it also ensures that performance of the whole Crucible instance is not impacted by a single process. The chart below shows the performance improvement we have seen from this modification.
Other announcements

- **Suggested reviews removed**
  As described above, we have removed the suggested reviews step during review creation, to improve Crucible performance and to ensure that review creation never interferes with the responsiveness of your Crucible instance.

- **Remote API setting always on**
  We have removed the Remote API setting from the Server Settings page. From this release onwards, Remote APIs are always accessible, which helps to make the JIRA integration straightforward.

- **Changes to the JIRA settings dialog**
  - *Allow Time Tracking Submission* is no longer a configurable option. Users are always able to submit time for their reviews when the JIRA integration is set up.
  - We have simplified permissions by removing the *Disable for Unmapped Projects* setting. All permissions to view the reviews tab are now controlled by the Project permission settings in JIRA.
  - *Force Project Mappings* has been removed as part of simplifying the JIRA FishEye Plugin behavior. Project links are respected, and act as restrictions on the integration.
  - *Max review age* is now only applied to versions of Crucible prior to 2.9.0.

- **Changes to the diff view in reviews**
  In order to facilitate the review process and give a clearer interface to see all the modifications and comments on a particular file we are now showing one file at a time in the reviews. You can use the file tree or the keyboard shortcuts to navigate between files.

The Crucible 2.9 team

Development

Core team

Brendan Humphreys
Conor MacNeill
Geoff Crain
Lukasz Pater
Maciej Swinarski
Piotr wity wicicki
Richard Stephens
Tom Davies
Valery Trubnikov

Team lead

Nick Pellow
Crucible 2.9 changelog

This page contains information about the Crucible 2.9 minor releases. FishEye license holders should also check the FishEye 2.9 Changelog. See the Crucible 2.9 release notes for details of what's new in 2.9.0.

Please read the Crucible 2.9 Upgrade Guide before upgrading to any of the minor releases below.

**From 2.9.1 to 2.9.2**

11 December 2012

This is a bug fix release. The complete list of issues is below.

<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 2.9.0 to 2.9.1

**19 November 2012**

This is a bug fix release. The complete list of issues is below.

<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 2.9

### 14 November 2012

The complete list of issues is below.

<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-6105</td>
<td>Chrome crashes</td>
<td>Maciej Swinarski</td>
<td>Renan Battaglini</td>
<td>Fixed</td>
<td>Apr 05, 2012</td>
<td>Oct 17, 2012</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
s if during a code review the user navigate using the shortcuts

CRUC-5781  In a Review containing a single Attachment, the 'Go to next element' button causes an infinite JavaScript loop

CRUC-5731  No entity exists with key ${helper.project.key} of type com.atlassian.applinks.api.application.fecru.FishEyeCrucibleProjectEntity

Crucible 2.9 upgrade guide

Please refer to the FishEye 2.9 upgrade guide for important notes on upgrading to Crucible 2.9.

For details of the new features and improvements in this release, please read the Crucible 2.9 Release Notes.

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!

## 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.
2 Shares

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.

3 Improved performance for the projects listing

We improved the rendering performance of the projects listing in Crucible.
4 Support for Subversion 1.7

Crucible now supports Subversion 1.7.

5 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

Product management
Sten Pittet

Project manager
Anton Mazkovoich
Support
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.

On this page:

- From 2.8.1 to 2.8.2
- From 2.8.0 to 2.8.1
- Crucible 2.8

From 2.8.1 to 2.8.2

5 October 2012

This is a FishEye bug fix release.

From 2.8.0 to 2.8.1

29 August 2012

This is a bug fix release. The complete list of issues is below.

<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>

Please read the Crucible 2.8 Upgrade Guide before upgrading to any of the minor releases below.
### Crucible 2.8

**15 August 2012**

The complete list of issues is below.

<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-6156</td>
<td>Test upgrade from 2.7 to 2.8 on all Supported DBs</td>
<td>Unassigned</td>
<td>Nick Pellow [Atlassian]</td>
<td>P</td>
<td>Clos ed</td>
<td>Fixed</td>
<td>Jun 21, 2012</td>
<td>Jul 02, 2012</td>
</tr>
</tbody>
</table>

Authenticate to retrieve your issues
**CRUC-6188**  
Crucible privilege escalation vulnerability  
Vitaly Osipov [Atlassian]  
Paul Watson [Atlassian]  
Closed  
Fixed  
Aug 10, 2012  
Aug 16, 2013

**CRUC-6168**  
A null review participant causes review indexing to fail partway through the set of reviews  
Tom Davies [Atlassian]  
Tom Davies [Atlassian]  
Closed  
Fixed  
Jul 10, 2012  
Sep 27, 2012

**CRUC-6166**  
Setup Func Tests are failing under JDK 1.7  
Geoff Crain [Atlassian]  
Nick Pellow [Atlassian]  
Closed  
Fixed  
Jul 06, 2012  
Jul 09, 2012

**CRUC-6162**  
Investigate why Crucible seems to auto-save always exactly when you go to post a comment.  
Nick Pellow [Atlassian]  
Nick Pellow [Atlassian]  
Closed  
Fixed  
Jul 02, 2012  
May 02, 2013

**CRUC-6157**  
Re-indexing crucible can cause a NullPointerException if LogItem table  
Richard Stephens [Atlassian]  
Richard Stephens [Atlassian]  
Closed  
Fixed  
Jun 22, 2012  
Jul 09, 2012
|-----------|---------------------------------------------------------------|------------------------|----------------------------|-------|---------|--------------|--------------|
## 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 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 refer to the Crucible upgrade guide.

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 there for known issues, 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

**Download latest version**

- Thank you for all your issues and votes. Keep logging issues to help us keep improving!
- Read the release notices for important information about this release.

**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.

More...

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>Type</th>
<th>Key</th>
<th>Summary</th>
<th>Assignee</th>
<th>Reporter</th>
<th>Priorit y</th>
<th>Status</th>
<th>Resolution</th>
<th>Created</th>
<th>Updated</th>
</tr>
</thead>
<tbody>
<tr>
<td>☑</td>
<td>FE-417</td>
<td>Getting</td>
<td>Unassi</td>
<td>David</td>
<td>↑</td>
<td>🌵 Clos ed</td>
<td>Fixed</td>
<td>Jul 05, 2012</td>
<td>Jul 09,</td>
</tr>
<tr>
<td>Id</td>
<td>Title</td>
<td>Status</td>
<td>Assigned To</td>
<td>Created</td>
<td>Fixed</td>
<td>Completed</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>
</tbody>
</table>
CRUC-6136
java.lang.IllegalArgumentException:
No local entity with key 'PROJECT
ECTNAME' and type 'class com.atlassi
assian.application.fecru.FishEyeCru
cilEntityTypeImpl' exists

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>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 2.7.12 to 2.7.13

This is a bug fix release. The complete list of issues is below.

<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>FE-410</td>
<td>FE-410 6</td>
<td>Fishey e 2.7.13 startup error when FISHE YE_IN ST have spaces</td>
<td>Nick Pellow</td>
<td>Leo Leung</td>
<td></td>
<td></td>
<td>Fixed</td>
<td>May 24, 2012</td>
<td>Jun 08, 2012</td>
</tr>
</tbody>
</table>

Authenticate to retrieve your issues

Created by Atlassian in 2013. Licensed under a Creative Commons Attribution 2.5 Australia License.
<table>
<thead>
<tr>
<th>Issue</th>
<th>Description</th>
<th>Status</th>
<th>Fixed</th>
<th>Created</th>
<th>Resolved</th>
</tr>
</thead>
<tbody>
<tr>
<td>CRUC-6092</td>
<td>Email generated via “Tools -&gt; Email Review” has incorrect comment URLs</td>
<td>Fixed</td>
<td>Mar 16, 2012</td>
<td>May 18, 2012</td>
<td></td>
</tr>
<tr>
<td>FE-4018</td>
<td>web-server/max-threads increases from 20 to 150</td>
<td>Fixed</td>
<td>Mar 14, 2012</td>
<td>Nov 18, 2012</td>
<td></td>
</tr>
<tr>
<td>FE-4017</td>
<td>EyeQL Search Results not displaying author fields</td>
<td>Fixed</td>
<td>Mar 27, 2012</td>
<td>Apr 10, 2012</td>
<td></td>
</tr>
<tr>
<td>FE-4017</td>
<td>Add authenticated user in http response headers for</td>
<td>Fixed</td>
<td>Mar 26, 2012</td>
<td>Apr 05, 2012</td>
<td></td>
</tr>
</tbody>
</table>

pre-2.7 behaviour: when a user is created by the user synchronisation, restore any deleted user with the same username.
<table>
<thead>
<tr>
<th>Issue</th>
<th>Description</th>
<th>Assigned to</th>
<th>Reporter</th>
<th>Assignee</th>
<th>Status</th>
<th>Resolution</th>
</tr>
</thead>
<tbody>
<tr>
<td>FE-398</td>
<td>Allow disabling of precise content Hash to csid mapping</td>
<td>Tom Davies (Atlassian)</td>
<td>Tom Davies (Atlassian)</td>
<td></td>
<td>Closed</td>
<td>Fixed</td>
</tr>
<tr>
<td>FE-392</td>
<td>Add custom syntax highlighting for shell scripts</td>
<td>Brenda Humphreys (Atlassian)</td>
<td>Eddie Webb</td>
<td></td>
<td>Closed</td>
<td>Fixed</td>
</tr>
<tr>
<td>FE-349</td>
<td>com.atlassian.crowd.exception.OperationFailedException if SSO is enabled with JIRA user management</td>
<td>Brenda Humphreys (Atlassian)</td>
<td>Gurleen Anand (Atlassian)</td>
<td></td>
<td>Closed</td>
<td>Fixed</td>
</tr>
<tr>
<td>CRUC-6075</td>
<td>Link to comments from dashboard</td>
<td>Richard Stephens (Atlassian)</td>
<td>Frederik Dinkela</td>
<td></td>
<td>Closed</td>
<td>Fixed</td>
</tr>
<tr>
<td>CRUC-6017</td>
<td>Allow plugins to detect when a draft comment is published</td>
<td>Richard Stephens (Atlassian)</td>
<td>Olli Nevalainen (Atlassian)</td>
<td></td>
<td>Closed</td>
<td>Fixed</td>
</tr>
<tr>
<td>CRUC-6009</td>
<td>Allow plugins</td>
<td>Brenda Humph</td>
<td>Olli Nevalainen</td>
<td></td>
<td>Closed</td>
<td>Fixed</td>
</tr>
<tr>
<td>Issue</td>
<td>Title</td>
<td>Assignee</td>
<td>Reporter</td>
<td>Status</td>
<td>Fixed</td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>----------------------------------------------------------------------</td>
<td>----------</td>
<td>----------</td>
<td>---------</td>
<td>----------------</td>
<td></td>
</tr>
<tr>
<td>FE-406</td>
<td>CC Indexing can fetch version 0 and index it even if before the start date.</td>
<td>Conor MacNeill</td>
<td>[Atlassian]</td>
<td>Conor MacNeill</td>
<td>Fixed, Closed, Apr 17, 2012, Apr 18, 2012</td>
<td></td>
</tr>
<tr>
<td>Ticket</td>
<td>Description</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>
</tr>
<tr>
<td>FE-404</td>
<td>Handle NPE caused by change set without a position value in Commit Graph</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FE-403</td>
<td>P4 GetLatestRevision will fail for paths with no revisions - leads to excessive logging</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FE-403</td>
<td>Copy from a file which existed before the start-revision is ignored</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FE-402</td>
<td>Store Perforce Branch Details not reflected in startup message of uninitializ ed</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Created by Atlassian in 2013. Licensed under a Creative Commons Attribution 2.5 Australia License.
### 16 April 2012

This is a bug fix release. The complete list of issues is below.

<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-400</td>
<td>Performance branch es are unnecessarily, repeatedly, scanned</td>
<td>Tom Davies [Atlassian]</td>
<td>Tom Davies</td>
<td>↓</td>
<td>Clos ed</td>
<td>Fixed</td>
<td>Mar 15, 2012</td>
<td>Apr 03, 2012</td>
</tr>
</tbody>
</table>
ct for projects including several paths in the same repository.

The 'Parameters' on the 'Database Configuration' page are ignored.

Authenticate to retrieve your issues

From 2.7.10 to 2.7.11

27 February 2012

This is a bug fix release. The complete list of issues is below.

<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>
Null Pointer Exception

Brendan Humphreys [Atlassian]

Cardinal Armand du Piessis de Richelieu, First Minister of Louis XIII

Content Indexing: Filter the paths to index before processing the list twice.

Matthew Watson [Atlassian]

Matthew Watson [Atlassian]

Lucene re-index is only triggered for cross-repo until after first commit

Matthew Watson [Atlassian]

Matthew Watson [Atlassian]

Content Indexing: Add an option to only index trunk, not root and trunk

Matthew Watson [Atlassian]

Matthew Watson [Atlassian]

ReIndex: Only delete all the

Matthew Watson [Atlassian]

Matthew Watson [Atlassian]
**User documentation for Crucible 3.1**

Content docs if there are >0 docs in the index

1. **CRUC-6051**  
   **Improve performance of the Review Blockers Report plugin**  
   Nick Pellow [Atlassian]  
   Nick Pellow [Atlassian]  
   Clos ed  
   Fixed  
   Feb 13, 2012  
   Feb 23, 2012

2. **FE-395**  
   **Per user LoC figures are meaningless when the 'All' button is selected**  
   Tom Davies [Atlassian]  
   Tom Davies [Atlassian]  
   Clos ed  
   Fixed  
   Feb 15, 2012  
   Feb 16, 2012

3. **FE-395**  
   **Breadcumb links on the Users tab are incorrect when a branch is selected**  
   Tom Davies [Atlassian]  
   Tom Davies [Atlassian]  
   Clos ed  
   Fixed  
   Feb 15, 2012  
   Feb 15, 2012

4. **FE-393**  
   **PHP syntax highlighting incorrectly matches keywords at the end of function names**  
   Brenda Humphreys [Atlassian]  
   Chris Butler  
   Clos ed  
   Fixed  
   Feb 08, 2012  
   Feb 08, 2012

---

*Created by Atlassian in 2013. Licensed under a Creative Commons Attribution 2.5 Australia License.*
|--------|----|----------------------------------------------------------|-----------------------------|-----------------------------|--------|-------|--------------|--------------|
| FE-392 | 1  | Fi shEye treats files as binary if they used to have a svn:mi
7 fires events for all commits and replays all smart commits

FE-388 5

Smart Commits: Using the #time command with a complex time format (i.e. 2d 4h) only saves the 2d part

Nick Pellow [Atlassian] Felipe Cuozzo [Atlassian]


FE-385 9

Unable to add users when using IE8

Seb Ruiz [Atlassian] Felipe Kraemer [Atlassian]

→ Closed Fixed Dec 12, 2011 Nov 08, 2012

FE-385 1

Old broken Support Zip link still present

Brendan Humphreys [Atlassian] Rene Verschoor [Atlassian]

→ Closed Fixed Dec 02, 2011 Feb 15, 2012

FE-360 8

Issue keys, wrapped in square brackets [JRA-9] are not linked correctly in quick search results

Brendan Humphreys [Atlassian] Nick Pellow [Atlassian]

→ Closed Fixed Jul 27, 2011 Feb 23, 2012

Authenticate to retrieve your issues
5 more issues

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>Type</th>
<th>Key</th>
<th>Summary</th>
<th>Assign</th>
<th>Report</th>
<th>Priorit</th>
<th>Status</th>
<th>Resolution</th>
<th>Create</th>
<th>Update</th>
</tr>
</thead>
<tbody>
<tr>
<td>📋</td>
<td>FE-390 3</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 [Atlassian]</td>
<td>▼</td>
<td>Clos ed</td>
<td>Fixed</td>
<td>Jan 19, 2012</td>
<td>Jan 19, 2012</td>
</tr>
<tr>
<td>📋</td>
<td>FE-390 0</td>
<td>fecru-review-issue-transition-plugin fails under fisheye 2.7.9 due to incompatable class change error</td>
<td>Unassigned</td>
<td>Nick Pellow [Atlassian]</td>
<td>▼</td>
<td>Clos ed</td>
<td>Fixed</td>
<td>Jan 19, 2012</td>
<td>Jan 19, 2012</td>
</tr>
</tbody>
</table>

Authenticate to retrieve your issues
### From 2.7.8 to 2.7.9

**19 January 2012**

This is a bug fix release. The complete list of issues is below.

<table>
<thead>
<tr>
<th>Type</th>
<th>Key</th>
<th>Summary</th>
<th>Assignee</th>
<th>Reporter</th>
<th>Priorit y</th>
<th>Status</th>
<th>Resolution</th>
<th>Create d</th>
<th>Update d</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 is starting up</td>
<td>Nick Pellow [Atlassian]</td>
<td>Alex Wei [Atlassian]</td>
<td></td>
<td>🌟 Clos ed</td>
<td>Fixed</td>
<td>Jan 05, 2012</td>
<td>Jan 18, 2012</td>
</tr>
</tbody>
</table>
FE-389 1  Webwork 2 vulnerability

FE-388 8  FishEye does not correctly handle moved tags

FE-388 4  Default UserManager. getUsersInGroup incorrectly acquire s a write lock instead of a read lock

FE-388 0  P4 client doesn’t handle job names with a leading -

FE-387 8  When processing performance file revisions, only branch specs that have been added or modified since the last restart are considered

Created by Atlassian in 2013. Licensed under a Creative Commons Attribution 2.5 Australia License.
<table>
<thead>
<tr>
<th>FE-387</th>
<th>5</th>
<th>SVN property changes with content lines containing certain strings break diff parsing</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>FE-386</th>
<th>2</th>
<th>JS errors in ie8</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>FE-386</th>
<th>1</th>
<th>lock management in Lucene Connection</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>FE-381</th>
<th>7</th>
<th>Clicking on repository in Fisheye Administration generates error dialog in IE8</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>FE-379</th>
<th>0</th>
<th>Trying to look at the source of an empty file results in java.lang.ArrayIndexOutOfBoundsException -1</th>
</tr>
</thead>
</table>

| CRUC-6028 | Internal avatar | Anna Buttfield | Brian McKenna | ↓  Clos  ed  Fixed  Jan 05, 2012  Jan 18, 2012 |

*Created by Atlassian in 2013. Licensed under a Creative Commons Attribution 2.5 Australia License.*
<table>
<thead>
<tr>
<th>CRUC-ID</th>
<th>Summary</th>
<th>Assigned To</th>
<th>Status</th>
<th>Fixed</th>
<th>Created</th>
<th>Updated</th>
</tr>
</thead>
<tbody>
<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>Anna Buttfield [Atlassian]</td>
<td>Fixed</td>
<td>Dec 28, 2011</td>
<td>Aug 28, 2012</td>
<td></td>
</tr>
<tr>
<td>CRUC-6010</td>
<td>Crucible search doesn't deal properly with author user names containing punctuation</td>
<td>Anna Buttfield [Atlassian]</td>
<td>Fixed</td>
<td>Dec 22, 2011</td>
<td>Jan 09, 2012</td>
<td></td>
</tr>
<tr>
<td>CRUC-6008</td>
<td>Stoped repositories cause NPEs in crucible reviews</td>
<td>Unassigned</td>
<td>Fixed</td>
<td>Dec 22, 2011</td>
<td>Feb 09, 2012</td>
<td></td>
</tr>
<tr>
<td>CRUC-6003</td>
<td>Timelines only work with a</td>
<td>Anna Buttfield [Atlassian]</td>
<td>Fixed</td>
<td>Dec 15, 2011</td>
<td>Jan 09, 2012</td>
<td></td>
</tr>
</tbody>
</table>
### 30 November 2011

This is a bug fix release. The complete list of issues is below.

<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>Create</th>
<th>Update</th>
</tr>
</thead>
<tbody>
<tr>
<td>Issue</td>
<td>Status</td>
<td>Assigned To</td>
<td>Created</td>
<td>Fixed</td>
<td>Description</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-384</td>
<td>Closed</td>
<td>Geoff Crain [Atlassian]</td>
<td>Nov 28, 2011</td>
<td>Nov 29, 2011</td>
<td>crowd admin groups get removed on startup</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FE-383</td>
<td>Closed</td>
<td>Geoff Crain [Atlassian]</td>
<td>Nov 21, 2011</td>
<td>Nov 22, 2011</td>
<td>autocomplete dropdowns give inconsistent and sometimes wrong results (results missing)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FE-377</td>
<td>Closed</td>
<td>Pierre-Etienne Poirot [Atlassian]</td>
<td>Oct 11, 2011</td>
<td>Nov 29, 2011</td>
<td>Allowed Review Participants field doesn't work when some users are present</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FE-374</td>
<td>Closed</td>
<td>Pierre-Etienne Poirot [Atlassian]</td>
<td>Sep 26, 2011</td>
<td>Nov 25, 2011</td>
<td>Changing Block Size triggers a reindex + restart, while only a restart is needed</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.

<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>

---

**Authenticate** to retrieve your issues
CRUC-1853

Display preferences menu doesn't disappear when filter menu clicked or menu item is selected


FE-383

In SVN, if a tag is created and the identified parent change set id is not indexed by FishEye the tag won't show up in the branches and tags dropdown or on the commit graph


FE-383

Performance jobids are not indexed


FE-383

Links in RSS feed do not use absolute URLs


FE-382

Jetty log messages

**FE-382**

**mime type mapping for file names rather than suffixes**

Tom Davies [Atlassian]

Tom Davies [Atlassian]

CLOSED  Fixed  Nov 09, 2011  Nov 17, 2011

**FE-317**

When navigating forward in time in activity stream, items are shown earliest to latest down the page

Michael Studman [Atlassian]

None

CLOSED  Fixed  Sep 22, 2010  Feb 01, 2012

**FE-315**

Browse /diff menu inactive looks active and vice-versa

Jonathan Poh [Atlassian]

None

CLOSED  Fixed  Dec 13, 2010  Nov 17, 2011

**CRUC-5977**

Crucible 'active users'

Tom Davies [Atlassian]

Tom Davies [Atlassian]

CLOSED  Fixed  Nov 15, 2011  Nov 17, 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.

<table>
<thead>
<tr>
<th>Type</th>
<th>Key</th>
<th>Summary</th>
<th>Assigned</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>----------------------------------</td>
<td>----------------------------------</td>
<td>--------</td>
<td>-------</td>
<td>---------------</td>
<td>---------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FE-361</td>
<td>Improv e the way avatars are loaded to avoid unnece ssary calls</td>
<td>Anna Buttfield [Atlassian]</td>
<td>Renan Battaglia n [Atlassian]</td>
<td>Closed</td>
<td>Fixed</td>
<td>Jul 29, 2011</td>
<td>Jan 23, 2013</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FE-382</td>
<td>If the total length of the branch names selecte d for the commit graph is too</td>
<td>Unassigned</td>
<td>Tom Davies [Atlassian]</td>
<td>Closed</td>
<td>Fixed</td>
<td>Nov 09, 2011</td>
<td>Nov 10, 2011</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
long, the spinner never disappears

FE-381 2 When synchronizing users with LDAP, do not deactivate users if the error is a communication problem

Pierre-Etienne Poirot [Atlassian]

Pierre-Etienne Poirot [Atlassian]

Closed Fixed Nov 02, 2011

Closed Fixed Nov 02, 2011

FE-380 9 Upgrade of instances using Oracle fail when upgrade_75.sql is run

Anna Buttfield [Atlassian]

Renan Battaglini [Atlassian]

Closed Fixed Oct 31, 2011

Closed Fixed Nov 02, 2011

FE-379 4 Sourc ing a Tag from a Tag Can cause FishEye not to display tag contents

Conor MacNeill [Atlassian]

Conor MacNeill [Atlassian]

Closed Fixed Oct 24, 2011

Closed Fixed Nov 03, 2011

FE-378 7 When synchronizing users with Crowd, do not deactivate users if the

Pierre-Etienne Poirot [Atlassian]

Adam Ahmed [Atlassian]

Closed Fixed Oct 20, 2011

Closed Fixed Aug 20, 2012
<table>
<thead>
<tr>
<th>Issue Key</th>
<th>Summary</th>
<th>Assignee(s)</th>
<th>Resolution</th>
<th>Created</th>
<th>Updated</th>
</tr>
</thead>
<tbody>
<tr>
<td>CRUC-5953</td>
<td>Scala syntax highlighter highlights substrings of identifiers as keywords</td>
<td>Seb Ruiz [Atlassian] Leif Wickland</td>
<td>Clos ed</td>
<td>Nov 01, 2011</td>
<td>Nov 02, 2011</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.

<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-376</td>
<td>FishEye logs lots of ApplicationPermissionException when synchronising Crowd users that have no permission on FishEye / Crucible</td>
<td>Michael Heemskerk [Atlassian]</td>
<td>Michael Heemskerk [Atlassian]</td>
<td>↓</td>
<td>Closed</td>
<td>Fixed</td>
<td>Oct 06, 2011</td>
<td>Oct 12, 2011</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>Title</td>
<td>Description</td>
<td>Author</td>
<td>Status</td>
<td>Date</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-369 2</td>
<td>When pushing to a Git managed repo while another client is cloning, the client which is cloned freezes</td>
<td>Pierre-Étienne Poirot [Atlassian]</td>
<td>Pierre-Étienne Poirot [Atlassian]</td>
<td>Fixed</td>
<td>Sep 06, 2011</td>
<td>Oct 09, 2011</td>
<td></td>
<td></td>
<td></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>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><img src="FE-374" alt="issue" /></td>
<td>FE-374 7</td>
<td>If you edit and then save a managed repository in Internet Explorer, the text</td>
<td>Tim Pettersen [Atlassian]</td>
<td>Tim Pettersen [Atlassian]</td>
<td>↑</td>
<td>Clos ed</td>
<td>Fixed</td>
<td>Sep 27, 2011</td>
<td>Sep 27, 2011</td>
</tr>
</tbody>
</table>
### FE-374 2

**After a user has been deleted and then added again, subsequent deletions fail**

<table>
<thead>
<tr>
<th>Created By</th>
<th>Status</th>
<th>Fixed By</th>
<th>Updated</th>
<th>Added By</th>
</tr>
</thead>
</table>

### FE-374 1

**Manual Auth Re-Syn c button doesn't do anything**

<table>
<thead>
<tr>
<th>Created By</th>
<th>Status</th>
<th>Fixed By</th>
<th>Updated</th>
<th>Added By</th>
</tr>
</thead>
</table>

### FE-373 8

**Log files not included in support zip**

<table>
<thead>
<tr>
<th>Created By</th>
<th>Status</th>
<th>Fixed By</th>
<th>Updated</th>
<th>Added By</th>
</tr>
</thead>
</table>

### FE-373 5

**Commit events fired on Git repos when branches are added/removed**

<table>
<thead>
<tr>
<th>Created By</th>
<th>Status</th>
<th>Fixed By</th>
<th>Updated</th>
<th>Added By</th>
</tr>
</thead>
</table>

### FE-373 4

**Error upon exceeding license limit**

<table>
<thead>
<tr>
<th>Created By</th>
<th>Status</th>
<th>Fixed By</th>
<th>Updated</th>
<th>Added By</th>
</tr>
</thead>
</table>

"fork of null" will appear and the forks view will no longer be availabl e.
FE-371
6
Hg authentication fails with special character in password
Clos ed Fixed Sep 14, 2011 Jan 17, 2012

FE-316
8
Restricting Fish eye content in Projects Has No Effect
Michael Studman [Atlassian]  None  
Clos ed Fixed Dec 06, 2010 Feb 01, 2013

CRUC-5920
ConstraintViolationException: Duplicate entry 'xyz' for key 'cru_user_name' after upgrading to 2.7
Michael Heemskerk [Atlassian]  Michael Heemskerk [Atlassian]  
Clos ed Fixed Sep 26, 2011 Sep 28, 2011

CRUC-5900
Showing only time for each change set (not from today) on Add Content to Review screen does not make sense
Seb Ruiz [Atlassian]  Wojciech Seliga  
Clos ed Fixed Sep 11, 2011 Oct 05, 2011
From 2.7.2 to 2.7.3

20 September 2011

This is a bug fix release. The complete list of issues is below.

<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.
<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-371</td>
<td>Eclipse Git integration (egit) fails to clone from managed Git repository</td>
<td>Unassigned</td>
<td>Tim Pettersen [Atlassian]</td>
<td>▼</td>
<td>Clos ed</td>
<td>Fixed</td>
<td>Sep 15, 2011</td>
<td>Sep 19, 2011</td>
</tr>
</tbody>
</table>
3

Ion numbers at bottom of Repositories screen overlap with system info footer

FE-3647

Google collections and Guava both bundled, package exported without version

CRUC-5902

Change set discussion autosave fails with error dialog when no comment is entered

CRUC-5888

Closing the Send Reminder dialog via Esc doesn't reset it.

CRUC-5886

If the current user is a review's author, then
switche
to a
reviewe
r, they
show
up in
the
manual
Send
Remind
er
dialog.

CRUC-
5885
Review
 remind
er
notes
do not
deren
new
lines in
e-mail

Michael
Studma
n
[Atlassi
an]

Seb
Ruiz
[Atlassi
an]

CRUC-
3298
Should
be able
to mark
files as
text
when
the
scm
reports
binary

Unassi
gned

Eduar
do M

Authenticate to retrieve your issues

From 2.7.0 to 2.7.1

9 September 2011

This is a bug fix release. The complete list of issues is below.

<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-370</td>
<td>Commit HookModuleDescription is not enabled can fail if Active Objects Configuration for Commit Hooks Plugin is not</td>
<td>Jason Hinch</td>
<td>Jason Hinch</td>
<td>↑</td>
<td>Clos ed</td>
<td>Fixed</td>
<td>Sep 08, 2011</td>
<td>Sep 12, 2011</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.

On this page:

- Upgrade Notes
  - Crucible 2.7
- Upgrade Procedure
- Checking for Known Issues and Troubleshooting the Crucible Upgrade
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:

  - **Crucible Plugin**
  - **Confluence Crucible Plugin**

- **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 refer to the **Crucible upgrade guide**.

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.

**RELATED TOPICS**

Crucible 2.7 Release Notes

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

Thank you for all your issues and votes. Keep logging issues to help us keep improving!
Read the release notices for important information about this release.

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.

2

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.

More...

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.

More...

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.

More...

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.
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.

<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>

 Authenticate to retrieve your issues

**From 2.6.5 to 2.6.6**

2 September 2011

This is a bug fix release. The complete list of issues is below.

<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>Create</th>
<th>Update</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></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.

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.
## From 2.6.2 to 2.6.3

### 9 August 2011

This is a bug fix release. The complete list of issues is below.

<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.

<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-5830</td>
<td>Admin allows the creation of permission schemes with blank names</td>
<td>Pierre-Etienne Poirot</td>
<td>Pierre-Etienne Poirot</td>
<td>🟢</td>
<td>Closed</td>
<td>Fixed</td>
<td>Jul 18, 2011</td>
<td>Jul 20, 2011</td>
</tr>
<tr>
<td>🟢</td>
<td>CRUC-5827</td>
<td>10 minute page load time for pre-commit diffs of large (1000 LOC) files</td>
<td>Geoff Crain</td>
<td>Nick Pellow</td>
<td>✰</td>
<td>Closed</td>
<td>Fixed</td>
<td>Jul 13, 2011</td>
<td>Jul 18, 2011</td>
</tr>
</tbody>
</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>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></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
CRUC-5809  Review s with anchor ed patche s cause NPE when anchor ed repos itory isn’t availabl e.

CRUC-5783  Applink s to non-JI RA servers confus e the Crucibl e link review to issue proces s

CRUC-5782  Excepti on while looking up JIRA issue ends request in error

CRUC-5780  NPE when trying to link review to JIRA issue causes failure of review creatio n

CRUC-5778  Moving a review betwee n project s fails when
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 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 refer to the Crucible upgrade guide.

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 2.5 Release Notes

8 February 2011

With great pleasure, Atlassian presents **Crucible 2.5**, now with support for Oracle and redesigned Activity Stream.

**Highlights of this Release:**

- Oracle Support (beta)
- Redesigned Activity Stream
- Improved Header
- Comment Notification Batching
- And Even More Improvements

**Responding to your Feedback:**

🌟 Over 150 votes satisfied

![Download latest version](download.png)

- Thank you for all your issues and votes. Keep
logging issues to help us keep improving!
- Read the release notices for important information about this release.

**Highlights of Crucible 2.5**

1

**Oracle Support (beta)**

We have been working very hard on adding support for the Oracle DBMS for Crucible 2.5 as this has been a very popular feature request with more that 115 votes.

We have been using Crucible with Oracle over the past month ourselves, so we believe we are almost there. We would love for you to give Crucible a try on Oracle DBMS and let us know if you run into any problems by creating an issue.

2

**Redesigned Activity Stream**

We have spent a considerable amount of time improving the FishEye activity stream:

- Cleaner visual design
- Larger avatars
- Improved scanability
- De-cluttering of UI elements, showing actions on hover
- Space-saving condensed mode so you can see more changesets on the same page

<table>
<thead>
<tr>
<th>Activity</th>
<th>Reviews</th>
<th>Favourites</th>
</tr>
</thead>
<tbody>
<tr>
<td>All</td>
<td>Commits</td>
<td>Reviews</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Issues</td>
</tr>
</tbody>
</table>

Today

- **Shankie**: CR-FE-4838 started review
  - CRUC-5565: Switched the added and removed block order in the code. Sorted.
  - 9 mins ago

- **Anna Battfield**: CR-FE-4819 summarized and closed
  - 17 mins ago

- **Shankie**: CR-FE-4857 started review
  - CRUC-5557: Strengthened the DOM order of the repository link in IE7
  - 18 mins ago

- **Anna Battfield**: CR-FE-4719 summarized and closed
  - 19 mins ago

- **Michael Stidman**: CR-FE-4771 summarized and closed
  - 5 hrs ago

3

**Improved Header**
Crucible 2.5 tracks locations that you have recently visited, and provides quick and easy access to navigate to these resources. The five most recently viewed projects, reviews, users and repositories (if FishEye is installed) are available from the header drop down links as shown:

Comment Notification Batching

Email notifications from Crucible can be frequent, particularly on reviews with many participants and lots of comments. We won’t say spam, but it can certainly feel “chatty!” Crucible 2.5 remedies this problem with improved batching of emails. All comments are grouped together and sent in a user-specified interval. They’re easier to read and will keep your inbox happy. It works for draft comments and edited comments, too, so fixing a typo will no longer send out multiple notifications!

Example:
There are 2 new comments in the following thread:
Geoff Crain on 02 Feb 2011, 12:01
   Is this really how this is supposed to work? You should fix this.
****** New
************************************************************************
***************************
Seb Ruiz on 02 Feb 2011, 13:31
   Geoff is right. But you'll have to fix it on a separate branch.
************************************************************************
****************************************
****** New
************************************************************************
**********************
Joe Xie on 02 Feb 2011, 13:32
   Fixed.. change is here:
   http://localhost:6060/crucible/changelog/FE?cs=e962e59bccda
************************************************************************
************************************
And Even More Improvements

- The **Universal Plugin Manager**
- Improved submit time dialog to JIRA:
Visit our issue tracker to see the full list of improvements and bug fixes in FishEye and Crucible for this release.

**Release Notices**

**Crucible 2.5 Changelog**

This page contains information about the Crucible 2.5 minor releases. FishEye license holders should also check the FishEye 2.5 Changelog.

> Please read the Crucible 2.5 Upgrade Guide before upgrading to any of the minor releases below.

**On this page:**
- From 2.5.6 to 2.5.7
- From 2.5.5 to 2.5.6
- From 2.5.4 to 2.5.5
- From 2.5.3 to 2.5.4
- From 2.5.2 to 2.5.3
- From 2.5.1 to 2.5.2
- From 2.5.0 to 2.5.1

**From 2.5.6 to 2.5.7**

**22 June 2011**

This is a bug fix release. The complete list of issues is below.

<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>

Please read the Crucible 2.5 Upgrade Guide before upgrading to any of the minor releases below.
CRUC-5778

Moving a review between projects fails when the default moderator of the destination project is the same as the current moderator.


CRUC-5773

List of JIRA Projects in linked server is cached forever.


CRUC-5707

Review coverage report gadget is broken.


From 2.5.5 to 2.5.6

24 May 2011

This is a bug fix release. The complete list of issues is below.

<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 2.5.4 to 2.5.5

11 May 2011

This is a bug fix release. The complete list of issues is below.

<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>Create</th>
<th>Updated</th>
</tr>
</thead>
</table>

Created by Atlassian in 2013. Licensed under a Creative Commons Attribution 2.5 Australia License.
### From 2.5.3 to 2.5.4

**11 April 2011**

This is a bug fix release. The complete list of issues is below.

<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-5715</td>
<td></td>
<td>After migrating to a different database,</td>
<td>Unassigned</td>
<td>Tom Davies</td>
<td></td>
<td>Closed</td>
<td>Fixed</td>
<td>Apr 06, 2011</td>
<td>Apr 07, 2011</td>
</tr>
</tbody>
</table>

**User documentation for Crucible 3.1**

Created by Atlassian in 2013. Licensed under a Creative Commons Attribution 2.5 Australia License.
session factories injected into spring beans still point to the old database.

CRUC-5710  Web Resources are loaded twice on pages rendered with some decorators

CRUC-5673  Time field displayed incorrectly when editing review

CRUC-5657  Add comment using Chrome ends up with no review content

CRUC-5644  RSS Feed option under Tools menu is not displaying correctly on IE7

CRUC-5585  Date format

Created by Atlassian in 2013. Licensed under a Creative Commons Attribution 2.5 Australia License.
error in edit crucible review dialog

CRUC-5427 reloading content in reviews doesn't work at all.


From 2.5.2 to 2.5.3

28 March 2011

This is a bug fix release. The complete list of issues is below.

<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-5662</td>
<td>Branch Order in branch selector drop down should be sorted better</td>
<td>Seb Ruiz [Atlassian]</td>
<td>Brydie McCoy [Atlassian]</td>
<td>Safe</td>
<td>🟢 Closed</td>
<td>Fixed</td>
<td>Mar 08, 2011</td>
<td>Mar 21, 2011</td>
</tr>
<tr>
<td></td>
<td>CRUC-5661</td>
<td>Searching with</td>
<td>Seb Ruiz</td>
<td>Brydie McCoy</td>
<td>Safe</td>
<td>🟢 Closed</td>
<td>Fixed</td>
<td>Mar 08, 2011</td>
<td>Mar 21, 2011</td>
</tr>
<tr>
<td>Type</td>
<td>Key</td>
<td>Summary</td>
<td>Assignee</td>
<td>Reporter</td>
<td>Priority</td>
<td>Status</td>
<td>Resolution</td>
<td>Created</td>
<td>Updated</td>
</tr>
<tr>
<td>------</td>
<td>-------</td>
<td>-------------------------------------------------------------------------</td>
<td>---------------------</td>
<td>--------------------</td>
<td>----------</td>
<td>--------</td>
<td>------------</td>
<td>----------------</td>
<td>----------------</td>
</tr>
</tbody>
</table>

**From 2.5.1 to 2.5.2**

**8 March 2011**

This is a bug fix release. The complete list of issues is below.
(and dependent) plugins from being disabled.

CRUC-5735  XSS vulnerability in Crucible Author Mapping
Vitaly Osipov [Atlassian]  Andrew Lui [Atlassian Technical Writer]
Closed  Fixed  Apr 21, 2011  May 31, 2011

CRUC-5734  XSS vulnerability in Crucible's Snippets
Vitaly Osipov [Atlassian]  Andrew Lui [Atlassian Technical Writer]
Closed  Fixed  Apr 21, 2011  May 31, 2011

CRUC-5652  After 2.5 upgrade indexing stops with error "Changeset comments messages may not be null or empty"
Closed  Fixed  Feb 16, 2011  Mar 07, 2011

CRUC-5645  Files that are added/moved should take up the whole width of the pane, even if side by side view is selected
Geoff Crain [Atlassian]  Joe Xie [Atlassian]
Closed  Fixed  Feb 25, 2011  Mar 02, 2011
<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-5621</td>
<td>Getting</td>
<td>Anna</td>
<td>Anna</td>
<td>Closed</td>
<td>Fixed</td>
<td>Feb 15, 2011</td>
<td>Apr 07, 2011</td>
</tr>
</tbody>
</table>

Created by Atlassian in 2013. Licensed under a Creative Commons Attribution 2.5 Australia License.
the list of projects via the api (either rest-service/projects-v1 or project Service.getAllProjects) fails with a LazyInitilaizatio
nException

CRUC-5616 Empty responses from p4 commands cause NullPointerExc
eption

CRUC-5615 java.lang.NoClassDefFound Error: org/tigris/subversion/javahl/SV
NCient thrown on startup

CRUC-5539 Problem with handling files with spaces

CRUC-5429 Error generating FishEye backup on Windo
ws
### 1 more issues

**From 2.5.0 to 2.5.1**

10 February 2011

This is a bug fix release. The complete list of issues is below.

<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><img src="" alt=" " /></td>
<td>CRUC-4984</td>
<td>&quot;that is unreviewed&quot; filter is broken</td>
<td>Adam Ahmed [Atlassian]</td>
<td>Agnes Ro [Atlassian]</td>
<td>↑</td>
<td>Clos ed</td>
<td>Fixed</td>
<td>Nov 18, 2010</td>
<td>Feb 16, 2011</td>
</tr>
<tr>
<td><img src="" alt=" " /></td>
<td>CRUC-4787</td>
<td>Patch reviews from mercurial diffs don't anchor to their repository</td>
<td>Geoff Crain [Atlassian]</td>
<td>Nicolas Venegas</td>
<td>↓</td>
<td>Clos ed</td>
<td>Fixed</td>
<td>Nov 03, 2010</td>
<td>Nov 15, 2011</td>
</tr>
</tbody>
</table>

1 more issues
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>CRUC-5597</td>
<td>The “Tabs” on the statistics panel of the “Project View” do not work in JIRA 4.2.3 and JIRA 4.3</td>
<td>Joe Xie [atlassian]</td>
<td>Brenden Bain [Atlassian]</td>
<td>Down</td>
<td>Closed</td>
<td>Fixed</td>
<td>Feb 06, 2011</td>
<td>Feb 14, 2011</td>
</tr>
<tr>
<td>CRUC-5594</td>
<td>Adding an AppLink to a “standalone Crucible” creates a FishEye configuration that is unusable.</td>
<td>Luis Miranda [Atlassian]</td>
<td>Brenden Bain [Atlassian]</td>
<td>Up</td>
<td>Closed</td>
<td>Fixed</td>
<td>Feb 04, 2011</td>
<td>Feb 14, 2011</td>
</tr>
</tbody>
</table>
Crucible 2.5 Upgrade Guide

Below are some important notes on upgrading to Crucible 2.5. For details of the new features and improvements in this release, please read the Crucible 2.5 Release Notes.
On this page:

- Upgrade Notes
- Upgrade Procedure
- Checking for Known Issues and Troubleshooting the Crucible Upgrade

Upgrade Notes

- **Crucible 2.5:**
  - This is only relevant when Crucible is run in conjunction with FishEye. This release will trigger an automatic upgrade of the metadata index the first time Crucible is started. Crucible is still usable while this task runs in the background, however the FishEye search functionality will not contain all results. A warning will appear in the UI until the upgrade has complete. Please also see How do I Avoid Long Reindex Times When I Upgrade?

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 refer to the Crucible upgrade guide.

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 2.5 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 2.5 Release Notes

Crucible FAQ

Created by Atlassian in 2013. Licensed under a Creative Commons Attribution 2.5 Australia License.
Crucible FAQ

Answers to frequently asked questions about configuring and using Crucible.

- Crucible Resources
- 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
- Contributing to the Crucible Documentation
  - Crucible Documentation in Other Languages

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.

Crucible Resources

Resources for Evaluators

- Free Trial
- Feature Tour

Resources for Administrators

- Crucible Knowledge Base
- Crucible FAQ
- Guide to Installing an Atlassian Integrated Suite
- The big list of Atlassian gadgets
**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.)

The prime motivation behind the metadata checking is to prevent the creation of reviews with a large number of files which have no effective content changes. Sometimes this sort of problem is seen when there is some wholesale SVN property change - e.g. setting the line endings on all files. In that respect, this is a performance setting as it avoids the creation of large reviews. The properties themselves are still stored and managed in FishEye. There is no major performance impact on including properties in reviews.

Disabling this functionality can be done 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 or if you want to disable FishEye enter Disabled in the license field. Crucible standalone will actually use a "light" mode of FishEye that comes pre-bundled 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.

- **No need to reconnect your repositories**
  Any repositories that you have currently defined in FishEye will not be visible in Crucible after removing FishEye (no more Source tab). The repositories however, will continue to update as usual without any intervention. You can add additional 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](https://confluence.atlassian.com/display/CRUCIBLE/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](https://confluence.atlassian.com/display/CRUCIBLE/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 Adding content to 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 FishEye documentation.
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, 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.

Report a Bug

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 for the stand-alone products and 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 Issue

If you find a security issue in the product, open an issue on https://jira.atlassian.com in the relevant project.

- Set the security level of the bug to 'Reporters and Developers'.
- Set the priority of the bug to 'Blocker'.
- Provide as much information on reproducing the bug as possible.

All communication about the security issue 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 cannot find the right project to file your issue in, email the details to security@atlassian.com.

When reporting a security vulnerability, please keep in mind the following:

We need a technical description that allows us to assess exploitability and impact of the issue.

- Provide steps to reproduce the issue, including any URLs or code involved.
- If you are reporting an XSS, your exploit should at least pop up an alert in the browser. It is much better if the XSS exploit shows user's authentication cookie.
- If you are reporting an SQL injection, we want to see the exploit extracting database data, not just producing an error message.
- HTTP request / response captures or simply packet captures are also very useful to us.

Please refrain from sending us links to non-Atlassian web sites, or reports in PDF / DOC / EXE files. Image files are ok. Make sure the bug is exploitable by someone other than the user himself (e.g. "self-XSS").

Without this information it is not possible to assess your report and it is unlikely to be addressed.

We are not looking for the reports listing generic "best practice" issues such as:

- Specific cookies being not marked as Secure or HTTPOnly
• Presence or absence of HTTP headers (X-Frame-Options, HSTS, CSP, nosniff and so on)
• Clickjacking
• Mixed HTTP and HTTPS content
• Auto-complete enabled or disabled
• SSL-related issues

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 systems).

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:

• Customer contact: We get the chance to meet customers and hear their successes and challenges at Atlassian Summit, Atlassian Unite, developer conferences, and road shows.
• Customer interviews: All product managers at Atlassian do customer interviews. Our interviews are not simply to capture a list of features, but to understand our customers' goals and plans.
• Community forums: There are large volumes of posts on answers, of votes and comments on jira.atlassian.com, and of conversations on community forums like groups on LinkedIn.
• Customer Support: Our support team provides clear insights into the issues that are challenging for customers, and which are generating the most calls to support
• Atlassian Experts: Our Experts provide insights into real-world customer deployments, especially for customers at scale.
• Evaluator Feedback: When someone new tries our products, we want to know what they liked and disliked and often reach out to them for more detail.
• In product feedback: The JIRA Issue Collectors that we embed our products for evaluators and our Early Access Program give us a constant pulse on how users are experiencing our product.
• Usage data: Are customers using the features we have developed?
• Product strategy: 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**

**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)
- 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](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 critical severity 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.
- We will send a copy of all posted 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](http://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.

If you want to track non-critical severity security vulnerabilities, you need to monitor the issue trackers for the relevant products on [http://jira.atlassian.com](http://jira.atlassian.com). For example, [https://jira.atlassian.com/browse/JRA](https://jira.atlassian.com/browse/JRA) for JIRA and [https://jira.atlassian.com/browse/CONF](https://jira.atlassian.com/browse/CONF) for Confluence. Security issues in trackers will be marked with a "security" label. All security issues will be listed in the release notes of the release where they have been fixed, similar to other bugs.

One of the ways to monitor updates to security issues is subscribing to the results of a sample search via email or RSS.

**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 OnDemand, since OnDemand 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
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>
<tr>
<td>3 – 5.9</td>
<td>Medium</td>
</tr>
<tr>
<td>6.0 – 7.9</td>
<td>High</td>
</tr>
<tr>
<td>8.0 – 10.0</td>
<td>Critical</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 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
  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 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).

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:

```
2009-07-16 16:56:12,390 ERROR [ThreadPool1] fisheye.app
com.cenqua.crucible.actions.admin.database.DBEditHelper doGet -
Database migration failed:
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
```

You may not see the message if you are running MySQL with default settings.

For more information, see the JIRA issue.

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.

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**

Author Guidelines
Atlassian Contributor License Agreement

**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

Glossary

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

- approve
- authors in Crucible
- code review
- comment
- creator
- defect
- moderator
- participant
- permission scheme
- permissions in Crucible
- projects in Crucible
- review duration
- reviewer
- role
- state
- statement of objective
users in Crucible

approve

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

authors in Crucible

The author is the person primarily responsible for acting on the outcomes of the review. In the vast majority of cases the author will be the person who made the code change under review.

Note: to map your repository username to your FishEye/Crucible username, see Changing your User Profile.

code review

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

See Getting Started.

comment

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

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.

See Creating a permission scheme.
permissions in Crucible

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

The following permissions are available:

<table>
<thead>
<tr>
<th>Permission</th>
<th>Description</th>
<th>Default Assignees</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abandon</td>
<td>Ability to abandon (i.e. cancel) a review.</td>
<td>Creator, Author, Moderator</td>
</tr>
<tr>
<td>Approve</td>
<td>Ability to approve a review (i.e. issue it 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>Creator, Author, Reviewer, Moderator</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>Complete</td>
<td>Ability of a reviewer to change their individual review status to Complete.</td>
<td>Reviewer</td>
</tr>
<tr>
<td>Create</td>
<td>Ability to create a review.</td>
<td>All logged-in users</td>
</tr>
<tr>
<td>Delete</td>
<td>Ability to delete a review.</td>
<td>Creator, Author, Moderator</td>
</tr>
<tr>
<td>Edit Review Details</td>
<td>Ability to edit a review's details and change the set of revisions being reviewed.</td>
<td>Creator, Author, Reviewer, Moderator</td>
</tr>
<tr>
<td>Re-Open</td>
<td>Ability to re-open a closed or abandoned review.</td>
<td>Creator, Author, Reviewer, Moderator</td>
</tr>
<tr>
<td>Recover</td>
<td>Ability to resurrect an abandoned (i.e. cancelled) review.</td>
<td>Creator, Author, Reviewer, Moderator</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>Creator, Author, 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, 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></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>
</tbody>
</table>
projects in Crucible

A Crucible project provides a way to group and manage related reviews – typically reviews that are all involved with the same software project. A Crucible 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 Editing 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.

role

See participant.

state

A Crucible review moves through the following states in the following sequence:

<table>
<thead>
<tr>
<th>State</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Draft</td>
<td>See Creating a Review.</td>
</tr>
<tr>
<td>Require Approval</td>
<td>Relevant only when the moderator is not the creator. See Issuing a Review.</td>
</tr>
<tr>
<td>Under Review</td>
<td>See Issuing a Review and Reviewing the Code.</td>
</tr>
<tr>
<td>Summarize</td>
<td>See Summarising and Closing the Review.</td>
</tr>
<tr>
<td>Closed</td>
<td>See Summarising and Closing the Review.</td>
</tr>
</tbody>
</table>

A review may also be in the following states:

<table>
<thead>
<tr>
<th>State</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abandoned</td>
<td>This happens when a review is deleted.</td>
</tr>
<tr>
<td>Rejected</td>
<td>Any reviews that a moderator has rejected.</td>
</tr>
<tr>
<td>Needs Fixing</td>
<td>This means that the review state is not understood by Crucible, and indicates a programming or data issue. The review moderator can move the review into a known state if this happens.</td>
</tr>
</tbody>
</table>

View | Ability to view a review. (People without this permission will not know that the review exists.) | Anonymous users All logged-in users Creator Author Reviewer Moderator |
statement of objective

A statement of objective is an optional text description of the review and any specific areas the reviewers should focus on.

users in Crucible

A user is a person using Crucible.