Table of Contents

Viewing Revision Logs 3
The Revision Log Form 3
The Revision Log Information 3
Inspecting Changes Between Revisions 4
Alternative Formats 4
The ChangeLog Text 4
RSS Support 5

Viewing Revision Logs

[TracGuideToc]

When you browse the repository, it's always possible to query the *Revision Log* view corresponding to the path you're currently seeing. This will display a list of the most recent changesets in which the current path or any other path below it has been modified.

The Revision Log Form

It's possible to set the revision at which the revision log should start, using the *View log starting at* field. An empty value or a value of *head* is taken to be the newest changeset.

It's also possible to specify the revision at which the log should stop, using the *back to* field. By default, it's left empty, which means the revision log will stop as soon as 100 revisions have been listed.

Also, there are three modes of operation of the revision log.

By default, the revision log *stops on copy*, which means that whenever an *Add*, *Copy* or *Rename* operation is detected, no older revision will be shown. That's very convenient when working with branches, as one only sees the history corresponding to what has been done on the branch.

It's also possible to indicate that one wants to see what happened before a *Copy* or *Rename* change, by selecting the *Follow copies* mode. This will cross all copies or renames changes. Each time the name of the path changes, there will be an additional indentation level. That way, the changes on the different paths are easily grouped together visually.

It's even possible to go past an *Add* change, in order to see if there has been a *Delete* change on that path, before that *Add*. This mode corresponds to the mode called *Show only adds, moves and deletes*. While quite useful at times, be aware that this operation is quite resource intensive.

Finally, there's also a checkbox *Show full log messages*, which controls whether the full content of the commit log message should be displayed for each change, or only a shortened version of it.

The Revision Log Information

For each revision log entry, there are 7 columns shown:

1. The first column contains a pair of radio buttons and should used

for selecting the *old* and the *new* revisions that will be

```
used for [[:TracRevisionLog#viewingtheactualchanges|viewing the actual
```

```
changes]].
```

2. A color code (similar to the one used for the changesets)

```
  indicating kind of change.
  Clicking on this column refreshes the revision log so that it restarts
  with this change.
```

3. The **Date** at which the change was made. 4. The **Revision** number, displayed as `@xyz`.

```
  This is a link to the TracBrowser, using that revision as the base line.
```

5. The **Changeset** number, displayed as `xyz`.

```
  This is a link to the TracChangeset view.
```

6. The **Author** of the change. 7. The **Log Message**, which contains either a summary or the full commit

```
  log message, depending on the value of the //Show full log messages//
  checkbox in the form above.
```

Inspecting Changes Between Revisions

The *View changes...* buttons (placed above and below the list of changes, on the left side) will show the set of differences corresponding to the aggregated changes starting from the *old* revision (first radio-button) to the *new* revision (second radio-button), in the TracChangeset view.

Note that the *old* revision doesn't need to be actually *older* than the *new* revision: it simply gives a base for the diff. It's therefore entirely possible to easily generate a *reverse diff*, for reverting what has been done in the given range of revisions.

Finally, if the two revisions are identical, the corresponding changeset will be shown (same effect as clicking on column 5).

Alternative Formats

The ChangeLog Text

At the bottom of the page, there's a *ChangeLog* link that will show the range of revisions as currently shown, but as a simple text, matching the usual conventions for ChangeLog files.

RSS Support

The revision log also provides a RSS feed to monitor the changes. To subscribe to a RSS feed for a file or directory, open its revision log in the browser and click the orange 'XML' icon at the bottom of the page. For more information on RSS support in Trac, see TracRss.

See also: TracBrowser, TracChangeset, TracGuide

From:
https://wiki.linux-ottawa.org/ - **Linux-Ottawa (OCLUG) Wiki**

Permanent link:
https://wiki.linux-ottawa.org/doku.php?id=tracrevisionlog

Last update: **2015/06/09 19:23**