

Facilities on FOSS

Dr Colin Turner Dr Gordon Crawford

University of Ulster

5th June 2007

Outline

- 1 Categories of User
- 2 Facility Overview

Categories of User

The website gives different facilities depending upon your involvement with projects.

Dependant on level of involvement

- **One time users**
Casual browsers, bug reporters and so on, there are certain disadvantages to not having a user account;
- **Logged in users**
Making a user brings many advantages, mainly an area that brings all your interests together, and (possibly) more access;
- **Project members**
Limited shell access, write access to source repositories;

Overview of Facilities

Our development environment, <http://foss.ulster.ac.uk>, has a number of facilities for users and developers:

- **Project pages and News** for general information and downloads;
- **Documentation / Recipes** which detail procedures and other information;
- **Support Tracker** allowing support questions to be asked, and answered;
- **Mailing Lists** for users, developers and so on;
- **Bug Trackers** allowing bugs and wishlists to be submitted and dealt with;
- **Task Trackers** allowing project planning.

Source Code

The source code is, of course, the centerpiece of the facilities. The source code is managed in a version control system, in this case **subversion**. This allows all changes and past history of the code to be tracked, and for old versions of the software to be managed. There are three ways to access the code.

Source Code

- 1 **Web Interface**: the option to “Browse” the source code gives access to a web interface that can be used to explore individual files, look at history, and download whole directories in **tar.gz** format. A UNIX standard, this can easily be read by good Windows utilities as well (e.g. WinZip).
- 2 **Anonymous SVN**: detailed on the website, this requires you to install an SVN client for your machine, and allows you to checkout the code. The advantage of this method is easy updates.
- 3 **Member SVN**: if you are a member of a project, you will be able to use your access to checkout code, and checkin changes.

Source Code

This raises a question. Although not everyone will have write access to the source code, we hope in the future several partners will join us and therefore that several people will. How can we check the source code is free from unfortunate or malicious changes from others?

We are implementing a mailing list for each project whose only job is to track changes made to the source code. This should be subscribed to by all developers on that project, so that they are up to speed with major changes, and everyone has visibility of all changes.

Source Code

Browsing the source code repositories reveals that they all have three directories in their root. These have certain traditional significance with subversion.

- **trunk** is the construction site - the place where most active development occurs. Generally this code might not work from one moment to the next and so it should **not** be run on production servers. It is for testing and development.
- **tags** contain links to specific versions, including stable ones. You should install from an appropriate tag when using a production environment.
- **branches** contains experimental deviations from the trunk (if any) that may be merged back in time.

Bug Tracking

Bug tracking is an important facility offered by FOSS. Bugs, and progress towards fixing them are left in the open, except in certain extreme cases where privacy is required. Both OPUS and the PDSsystem contain links to the bug tracker on FOSS at the bottom of their pages. You have (of course) the right to change this, but we ask that you continue to leave these links so all bugs are handled in one location.

The fields offered and required are configurable from one project to another, and so may differ, but some simple rules are:

Bug Tracking

Submitting a bug

- please check to see if a bug has already been filed before filing a new one;
- please distinguish between real bugs (problems with existing functionality), and wishlist items, that are really descriptions of features you would like (but these are filed on-line);
- please try to include as much detail as possible in your report, any fix will be delayed if we need to keep asking you for further information;
- please include your email address, or better yet, make a user on FOSS and log-in before submission.

Bug Tracking

What next?

- If you have submitted an email address, or are a logged in email, you will be CC'd on all correspondence on that issue, although you can switch this off at any time. It is also possible to list other parties who should be included in any correspondence.
- The bug will be “triaged”, that is, the development time will assign it a severity or priority, or adjust your initial settings. This indicated their judgement on what order bugs and items should be tackled.
- Logged in users can vote for bugs they regard as an urgent priority to assist developers in triage, or adjusting initial triage assessments.

Bug Tracking

- You can use a simple wiki markup when you are submitting or augmenting bug reports.
- As developers or others work through the bug, triaging it, obtaining more information and so on, all this information will be here, right up to when the bug is closed (and maybe even reopened if the fix is not successful).
- If you have a proposed fix for the problem, and don't have member access to effect it yourself, then you can attach a patch or further information.

Support Tracking

Support tracking works in an extremely similar way to Bug tracking, except that it is intended to allow questions to be asked, and answered, in one common place.

Task Tracking

Again, tracking tasks works in a similar way to other trackers. The task tracker is intended to coordinate the project, assign priorities and timelines to various issues.

It can also be used (like other trackers in fact) to map the dependencies of tasks to other tasks in the same or other projects, and upon bugs in the various trackers.

The Rest

Each project has a simple news service that allows news stories to be submitted, published, and commented upon. Mailing lists allow an interface to create mailing lists, view public archives and so on.

The Docs allow wiki like articles to be added with specific audiences.