

Getting Perl modules into Debian

Debian's Perl team from an end-user perspective

Tim Retout

11th September 2010
HantsLUG @ IBM Hursley



What is the Debian Perl team?

- Maintain almost 1700 Perl packages within Debian
- Both official Debian Developers and sponsees
- Around 70-80 committers,¹ but most activity from a small core team
- Also indirectly contributes most of the Perl packages in Debian's derivatives (e.g. Ubuntu).

¹<http://www.ohloh.net/p/pkg-perl>



How do you install Perl modules?

- `apt-get install libfoo-perl`
- Backports for older releases
- CPAN for unpackaged modules

CPAN (the tool) is configured to play nicely with Debian packages, but does not install packages when satisfying dependencies.



When a module isn't packaged

If the Perl module you need is not available, you have several options:

- 1 Install it with CPAN, and handle future upgrades yourself.
- 2 File an RFP (Request for package) bug in Debian.²
- 3 Build yourself a private Debian package with dh-make-perl, and handle future upgrades yourself.
- 4 Get involved and contribute it back to Debian!

Getting involved is easier than you think.

²<http://pkg-perl.alioth.debian.org/howto/RFP.html>



The Perl team uses tools to speed up packaging.

- alioth.debian.org for team management
- SVN for most packages, but a few are stored in git
- <http://bugs.debian.org/>
- Package Entropy Tracker (PET)
- IRC ([#debian-perl](#) on OFTC)
- Two mailing lists - one for discussion, one to receive automated messages

<http://pkg-perl.alioth.debian.org/> links to all of these.



Getting involved - joining the team

- Create an alioth account
- Apply to join the pkg-perl group on alioth
- Introduce yourself on the debian-perl mailing list
- Optionally use IRC for realtime communication

<http://wiki.debian.org/Teams/DebianPerlGroup/Welcome>



Creating a package - dh-make-perl

dh-make-perl can jump-start creating a new Perl package.

- `apt-get install dh-make-perl` and read the man page.
- Run `dh-make-perl --cpan Module::Name`.
- Add `--pkg-perl` if you are packaging for the team.
- Then run `debuild` and hack until it builds.

If you don't want to contribute it to Debian, you could stop here.



Creating a package - svn-buildpackage

Once you have an initial package, you can upload to SVN. pkg-perl uses a fairly complex SVN layout.³

- Use `svn-inject` to inject new packages
- `svn-buildpackage` for building
- `svn-upgrade` for upstream upgrades

Set the suite to 'UNRELEASED' in `debian/changelog` for now. Commits are logged to IRC.

³<http://pkg-perl.alioth.debian.org/subversion.html>



Creating a package - cowbuilder

One of the more tricky parts of packaging is getting the build dependencies correct. cowbuilder is a tool for building packages in a clean chroot:

- `sudo apt-get install cowbuilder`
- `sudo cowbuilder --create`
- `sudo cowbuilder --build libfoo-perl_1.0-1.dsc`
- Can also integrate it into `svn-buildpackage`.

Check the test suite gets run!



Creating a package - lintian

We want Debian packages to be of high quality. lintian is a tool that can check for common problems.

- `sudo apt-get install lintian`
- `lintian -iI --pedantic --color=auto libfoo-perl_1.0-1_amd64.changes`
- Can also integrate it with `debuild/cowbuilder`



Creating a package - getting feedback

Once you think your package is ready for review, mark it as such:

- `dch -r`
- (or 's/UNRELEASED/unstable/' in `debian/changelog`)
- `svn commit -m 'dch -r; ready for review'`

The package status will be changed automatically in PET.



Creating a package - what next?

- You are likely to get feedback via IRC or email.
- Sometimes TODO lists are left for you in debian/changelog by the reviewer, with the suite set back to UNRELEASED.
- Once the reviewer is happy, they will upload the package and tag the release in SVN.



Questions?

`<diocles@debian.org>`

