



The Pod Indexing Project

Ivan Tubert-Brohman
itub@cpan.org

One common problem with using the perl documentation is figuring out where to look for something (is it in perl, perlre, perlfunc, ...?). While it is easy to search for a function in perlfunc using `perldoc -f`, or to search questions from the perlfaq using `perldoc -q`, if one wants to search for variables, operators, random topics, and so on. There's really no easy way to do it. There is `perltoc`. `pod`, but it's huge and usually not very helpful for this purpose.

Some sites such as search.cpan.org and perldoc.perl.org provide search engines for the perl documentation. There's also a program called `perlindex` that comes with the Text-English distribution for searching installed Pods. However, search engines tend to have a couple of limitations:

- 1) they usually don't handle what some people call "line noise" very well. That is, searching for operators or special variables, such as `$*`.
- 2) they tend to return too many results and have no way of telling which place has a good "canonical definition" for a term.

The Pod Indexing Project addresses this problem by creating a hand-curated index to Perl's core documentation. It is based on the little-known (or at least little-used) `x<>` code, which is described in `perlpod`:

```
x<topic name> -- an index entry This is ignored by most formatters, but some may use it for building indexes. It always renders as empty-string. Example: x<absolutizing relative URLs>
```

Currently `x<>` is used in only one place in the perl documentation: `pod/perlfunc.pod` uses it for the "-X" file test operators.

The Pod Indexing Project will add relevant `x<>` tags throughout Perl's core documentation. They will all

be added manually, so the project needs volunteers. Just consider that perl ships with about 100 pod files, totalling over 100,000 lines. And that's just for the core documents (*i.e.*, the ones that are called `perl*`, such as `perlsyn`, `perl`, and `pervar`). Once those are done, we can start worrying about indexing the documentation for the modules that ship with the perl distribution (such as `CGI` and `Data::Dumper`)

Besides indexing the pods, I've been working on a module called `Pod::Index`, for generating and searching the index, and on a modified version of `perldoc` that supports a `-k` option for keyword search. Just typing something like

```
perldoc -k context
```

will display excerpts from all relevant documents that were tagged with `x<context>`. I've set up an online demo for searching the current (but incomplete) index at <http://pod-indexing.annocpan.org/perldoc-k.cgi>.

Getting the Sources

All patches we provide should be against the latest development perl, also known as `bleadperl` or `perl-current`. You can read the `perlhack` documentation page for more details. It explains how Perl development actually happens and where everything is. Here are the basics you need for this project, though.

You can browse `perl-current` at its ActiveState repository listed in the "References" section.

If you just want to get one file at a time, that's the easiest way, and may be enough for our documentation purposes.

If you want to mirror the whole thing, the recommended way is to use `rsync`:

```
$ rsync -avz rsync://ftp.linux.activestate.com/perl-current/ perl-current
```