

Perl and Its Community

One of Larry's main goals for Perl 5 was to encourage Perl development and evolution outside the core distribution. Perl 4 had several forks, because there was no easy way to connect it to a relational database, for example. Larry wanted people to create and maintain their own extensions without fragmenting Perl into thousands of incompatible pidgins.

You can add technical mechanisms for extensions, but you must also consider community aspects as well. Extensions and enhancements that no one shares are extensions and enhancements that everyone has to build and test and debug and maintain themselves. Yet shared extensions and libraries are worthless if you can't find them, or you can't enhance them, or you don't have permission to use them.

Fortunately, the Perl community exists. It's strong and healthy. It welcomes willing participants at all levels--and not just for people who produce and share code. Consider taking advantage of the knowledge and experience of countless other Perl programmers, and to share your abilities as well.

Community Sites

Perl's homepage is <http://www.perl.org/>. This site hosts documentation, source code, tutorials, mailing lists, and several important community projects. If you're new to Perl, the Perl beginners mailing list is a friendly place to ask novice questions and get accurate and helpful answers. See <http://beginners.perl.org/>.

An important subdomain of note is <http://dev.perl.org/>, a central site for core development of Perl 5, Perl 6. The main Perl 6 site is <http://www.perl6.org/>, and even Perl 1.

Perl.com publishes several articles and tutorials about Perl programming every month. Its archives reach back into the 20th century. See <http://www.perl.com/>.

The CPAN's (*cpan*) central location is <http://www.cpan.org/>, though experienced users spend more time on <http://search.cpan.org/>. This central software distribution hub of reusable, free Perl code is an essential part of the Perl community.

PerlMonks, at <http://perlmonks.org/>, is a venerable community site devoted to questions and answers and other discussions about Perl programming. It celebrated its tenth anniversary in December 2009, making it one of the longest lasting web communities dedicated to any programming language.

Several community sites offer news and commentary. The oldest is <http://use.perl.org/>; it has offered custom journals for many years. Several well-known Perl hackers write there. <http://blogs.perl.org/> is a newer community site dedicated to blogging about Perl.

Other sites aggregate the Perl-related musings of Perl hackers, including <http://perlsphere.net/>, <http://planet.perl.org/>, and <http://ironman.enlightenedperl.org/>. The latter is part of an initiative from the Enlightened Perl Organization (<http://enlightenedperl.org/>) to increase the amount of and improve the quality of Perl publishing on the web.

Perl Buzz (<http://perlbuzz.com/>) collects and republishes some of the most interesting and useful Perl news on a regular basis.

Development Sites

Best Practical Solutions (<http://bestpractical.com/>) maintains an installation of their popular request tracking system, RT, for CPAN authors as well as Perl 5 and Perl 6 development. Every CPAN distribution has its own RT queue, linked from `search.cpan.org` and available on <http://rt.cpan.org/>. Perl 5 and Perl 6 have separate RT queues available on <http://rt.perl.org/>.

The Perl 5 Porters (also known as *p5p*) mailing list is the central point for Perl 5 development discussion. See <http://lists.cpan.org/showlist.cgi?name=perl5-porters>.

The Perl Foundation (<http://www.perlfoundation.org/>) hosts a wiki for all things Perl 5. See <http://www.perlfoundation.org/perl5>.

Many Perl hackers use Github (<http://github.com/>) to host their projects... including the drafts of this book at http://github.com/chromatic/modern_perl_book/). One particularly notable page on Github is Gitpan, which hosts Git repositories chronicling the complete history of every distribution on the CPAN. See <http://github.com/gitpan/>.

Events

There are plenty of events in the physical world as well. The Perl community holds a lot of conferences, workshops, and seminars. In particular, the community-run YAPC--Yet Another Perl Conference--is a successful, local, low-cost conference model held on multiple continents. See <http://yapc.org/>.

The Perl Foundation wiki lists other events at http://www.perlfoundation.org/perl5/index.cgi?perl_events.

There are also hundreds of local Perl Mongers groups which get together frequently for technical talks and social interaction. See <http://www.pm.org/>.

IRC

When Perl mongers aren't at local meetups or conferences or workshops, many collaborate and chat online through IRC, a textual group chat system from the early days of the Internet. Many of the most popular and useful Perl projects have their own IRC channels, such as *#moose* or *#catalyst*.

The main server for Perl community is <irc://irc.perl.org/>. Other notable channels include *#perl-help*, for general assistance on Perl programming, and *#perl-qa*, devoted to testing and other quality issues. Be aware that the channel *#perl* is *not* for general help--instead, it's a general purpose room for discussing whatever its participants want to discuss... and it's not often friendly to people who ask basic programming questions..

The CPAN

Perl 5 is a pragmatic language. It'll help you get your work done. Yet the ever-pragmatic Perl community has extended that language and made their work available to the world. If you have a problem to solve, chances are someone's already uploaded code to the CPAN for it.

The line between a modern language and its libraries is fuzzy. Is a language only syntax? Is it the core libraries? Is it the availability of external libraries and the ease at which you can use them within

your own projects?

Regardless of how you answer those questions for any other language, modern Perl programming makes heavy use of the CPAN (<http://www.cpan.org/>). The CPAN, or Comprehensive Perl Archive Network, is an uploading and mirroring system for redistributable, reusable Perl code. It's one of--if not *the*--largest archives of libraries of code in the world.

CPAN mirrors *distributions*, which tend to be collections of reusable Perl code. A single distribution can contain one or more *modules*, or self-contained libraries of Perl code. Each distribution lives in its own namespace on the CPAN and contains its own metadata. You can build, install, test, and update each distribution. Distributions may depend on other distributions. For this reason, installing distributions through a CPAN client is often simpler than doing so manually.

The CPAN *adds* hundreds of registered contributors and thousands of indexed modules in hundreds of distributions every month. Those numbers do not take into account updates. In late April 2010, search.cpan.org reported 8156 uploaders, 81436 modules, and 20065 distributions.

The CPAN itself is merely a mirroring service. Authors upload distributions containing modules, and the CPAN sends them to mirror sites, from which users and CPAN clients download, configure, build, test, and install distributions. Yet the CPAN has succeeded because of this simplicity, and because of the contributions of thousands of volunteers who've built on this distribution system to produce something greater. In particular, community standards have evolved to identify the attributes and characteristics of well-formed CPAN distributions. These include:

Standards for installation to work with automated CPAN installers.

Standards for metadata to describe what each distribution includes and any dependencies of the distribution.

Standards for documentation and licensing to describe what the distribution does and how you may use it.

Additional CPAN services provide comprehensive automated testing and reporting of every CPAN distribution for adherence to packaging and distribution guidelines and correctness of behavior on various platforms and versions of Perl. Every CPAN distribution has its own ticket queue on <http://rt.cpan.org/> for reporting bugs and working with authors. Distributions also have historical versions available on the CPAN, ratings, annotations for the documentation, and other useful information. All of this is available from <http://search.cpan.org/>.

Modern Perl installations include two clients to connect to, search, download, build, test, and install CPAN distributions, CPAN.pm and CPANPLUS. They behave equivalently; their use is a matter of taste. This book recommends the use of CPAN.pm solely due to its ubiquity.

If you use a recent version of CPAN.pm (as of this writing, 1.9402 is the latest stable release), CPAN.pm configuration is largely decision-free. For any complete installation of Perl, you may start the client with:

```
$ B<cpan>
```

To install a distribution:

```
$ B<cpan Modern::Perl>
```

Eric Wilhelm's excellent tutorial on configuring CPAN.pm (

<http://learnperl.scratchcomputing.com/tutorials/configuration/>) is worth reading twice, especially its section on troubleshooting.

Even though the CPAN client is a core module for the Perl 5 distribution, you may also have to install standard development tools such as a `make` utility and possibly a C compiler to install all of the distributions you want. Windows users, see Strawberry Perl (<http://strawberryperl.com/>) and Strawberry Perl Professional. Mac OS X users need their developer tools installed. Unix and Unix-like users, consult your local system administrator.

For your work setting up a CPAN client and an environment to build and install distributions, you get access to libraries for everything from database access to profiling tools to protocols for almost every network device ever created to sound and graphics libraries and wrappers for shared libraries on your system.

Modern Perl without the CPAN is just another language. Modern Perl with the CPAN is amazing.

Serious Perl developers often manage their own Perl library paths or even full installations. Several projects help to make this possible.

`App::cpanminus` is a new CPAN client with goals of speed, simplicity, and zero configuration. Installation is as easy as:

```
$ curl -LO http://xrl.us/cpanm
$ chmod +x cpanm
```

`App::perlbrew` is a system to manage and to switch between your own installations of multiple versions and configurations of Perl. Installation is as easy as:

```
$ curl -LO http://xrl.us/perlbrew
$ chmod +x perlbrew
$ ./perlbrew install
$ perldoc App::perlbrew
```

The `local::lib` CPAN distribution allows you to install and to manage distributions in your own user directory, rather than for the system as a whole. This is an effective way to maintain CPAN distributions without affecting other users. Installation is somewhat more involved than the previous two distributions. See <http://search.cpan.org/perldoc?local::lib> for more details.

All three distributions projects tend to assume a Unix-like environment (such as a GNU/Linux distribution or even Mac OS X). Windows users, see the Padre all-in-one download (<http://padre.perlide.org/download.html>).