



Guidelines For Contributing To CPAN

Presented to <http://oc.pm.org/> on 2016-06-27
Erick Jordan (erickfjordan@gmail.com)

Comprehensive Perl Archive Network

CPAN's main purpose is to help programmers locate modules and programs not included in the Perl standard distribution.

According to <http://www.cpan.org/> CPAN currently has 169,627 modules in 33,895 distributions, written by 12,718 authors, mirrored on 241 servers. The archive has been online since October 1995 and is constantly growing.

How to Contribute

The home page for www.cpan.org has a section entitled “How to Contribute”. You are encouraged to take the time to read the section.

For the basics, there are two items needed to contribute to CPAN: 1) a PAUSE account 2) a Perl module. PAUSE stands for “Perl Authors Upload Server”. Your entry point to PAUSE can be found at pause.perl.org.

Naming Your Module

This is perhaps the most difficult step in the process.

Below are a few guidelines ...

- Name your module so that it's easy to find.
- The name should reflect what it does, what task is it doing?
- The name should fit with everything else in CPAN, what is the namespace?
- Avoid top-level namespaces (e.g. "HTML", "DB", "Net")
- Avoid Simple, Easy, Reduced, Tiny, Fast, Small, Super, Smart
- Avoid Devel, Sys, Text, Data

Naming Your Module cont'd

Simply put, identify what your module does and look for an existing namespace to fit it in. Example, `HTML::Parser`, `HTML::TreeBuilder` gives the reader an inkling of what it does. Whereas `HTML::Sys`, leaves him clueless.

For more details, you are encourage to read section “On the Naming of Modules” at the PAUSE website.

Preparing Your Module for Distribution

Preparing your module involves organizing the structure (e.g. subfolders, examples, test scenarios), writing build/make scripts, and bundling. You are encouraged to read <http://perldoc.perl.org/perlnewmod.html>. They offer useful tips as well as tools to create a Perl module template.

Some key points from the read ...

- Making the module using Module::Starter
- Use plain old documentation (POD) to document your module
- Writing tests to self-test your module
- Writing the README

Use GitHub

Once you've prepared your module you'll want to upload it to GitHub. You'll find everything runs smoother if you have a GitHub link to your project (or Perl module). For example, PrePAN (discussed in the next slide) will ask you for a GitHub link to review your module.

PrePAN

If you have never uploaded a module to CPAN before you are strongly encouraged to get feedback from prepan.org. PrePAN will help you answer questions such as ...

- Are there similar modules to my module?
- How can I distribute the module in keeping with CPAN's conventions?
- Does my module name make sense?
- Who can review my module?

Uploading Your Module To PAUSE

Login to your PAUSE account. Select Upload a file to CPAN -> Upload Material -> Choose File. Example, Parse-STDF-0.2.5.tar.gz is the tar bundle for module Parse::STDF version 0.2.5.

PAUSE will automatically send you an email as to the status of your module. It may take a few hours or so to complete. Once your module has been uploaded, you cannot change code without making a new version and hence a new tar bundle. You can however change READMEs and text files.

References

<https://cpan.org>

<https://pause.perl.org>

<https://prepan.org>

<https://github.com>

<http://perldoc.perl.org/perlnewmod.html>



)

Thank You.