Grokking The CPAN

Elaine Ashton, O’Reilly Open Source Convention, San Diego
Thursday, 26 July 2001

I. Genesis

What is The CPAN?

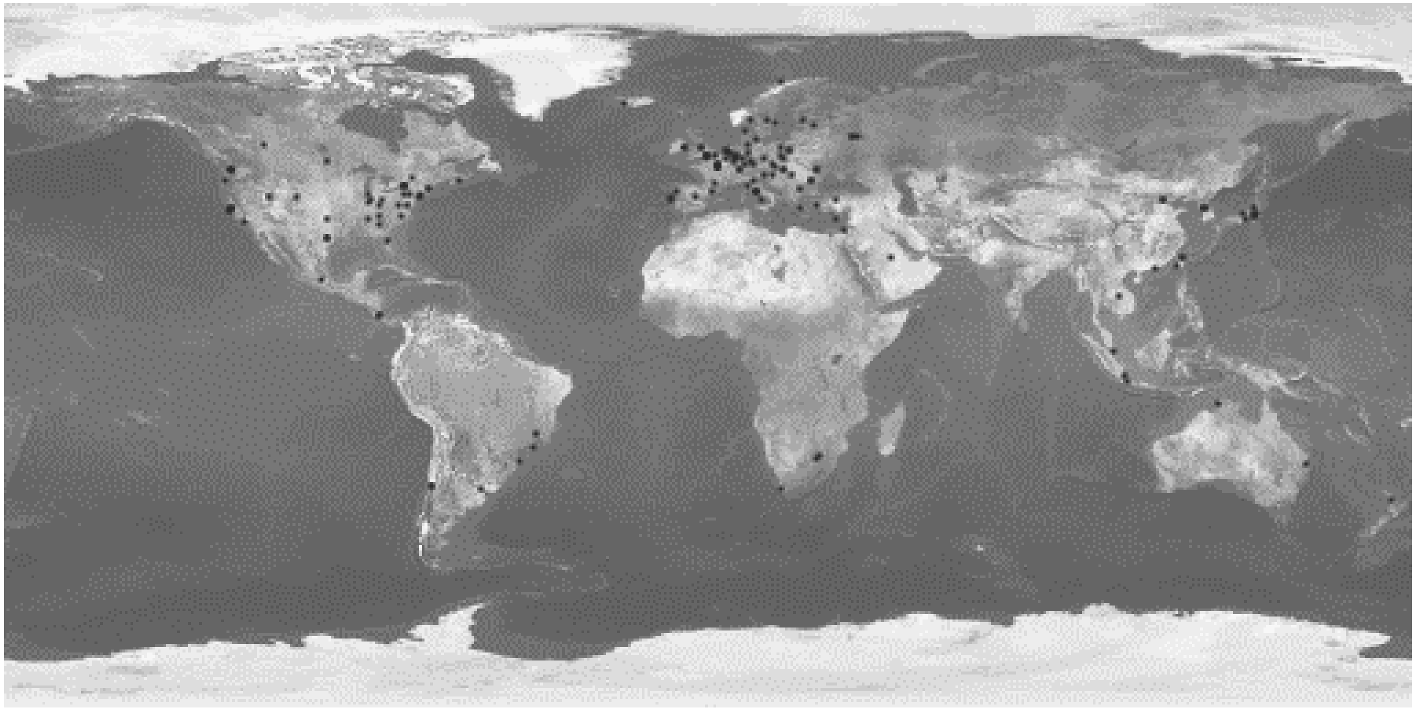
- Comprehensive Perl Archive Network - ~800mb and 180 mirrors
- Began as an idea by Jared Rhine in December 1993 as a place to store all the Perl bits then scattered all over the map.

"I congratulate on your effort to create a new Perl archive site. I too, have begun to create a comprehensive Perl archive. It is this about which I wish to speak. As you no doubt have noticed, there are far too many different structures of Perl archives out there and none are canonical. I propose that we cooperate to create a unified structure, much like the CTAN project which has managed to create a collection of canonical sites for TeX
[…]
The advantages to the Perl community would be numerous, I imagine. There would be no searching for the correct random directory to find something, and users would not have to go to multiple sites looking for the particular bit they are looking for."[http://history.perl.org/packratsarch/93 for the full message]

What is The CPAN contd.

- The idea sits dormant for over a year until Jarkko prods the packrats about it in early 1995.
- Jarkko works on the CPAN archive and Andreas also tosses around the concept of a ‘Master Module Archive’.
- CPAN announced to the world on 26 October 1995 http://www.cpan.org/clpa/1995-10/19951026-142858
- Who was instrumental in bringing CPAN from concept to reality?
    - Bill Middleton, Mark of Coombs, Tom Christiansen, Andreas Koenig, Jared Rhine, Jarkko Hietaniemi, Stephen Potter, Tim Bunce, Jon Orwant

Where is The CPAN?



180+ CPAN mirrors around the world.

http://mirror.cpan.org/

What is PAUSE and what does it have to do with The CPAN?

- PAUSE is the Perl Authors Upload Server - http://pause.cpan.org/
- PAUSE is run by Andreas Koenig
- PAUSE is represented by authors/ modules/ and scripts/ on the CPAN.

CPAN is often used when people speak of what really is PAUSE. PAUSE is the place where an author will upload a distribution, documentation or a script. The PAUSE will then extract a README if there is one, make the appropriate links, and create an MD5 checksum. CPAN then mirrors PAUSE once an hour and, from there, to the rest of the mirrors around the world.

II. Navigating

How is the CPAN organized?

- **CPAN.html -** General introduction to CPAN and it's resources.
- **ENDINGS** - What file extensions like .zip and .gz and .tar mean.
- **MIRRORED.BY -** Detailed information about all current CPAN mirrors.
- **MIRRORING.FROM** - What sites are currently mirroring content into the CPAN.
- **README** and **README.html** - [deprecated] A brief introduction to CPAN.
- **RECENT** and **RECENT.html** - What had been added in the last 7 days on CPAN.
- **SITES and SITES.html** - Long list of CPAN Mirrors listed by continent.
- **authors/ -** modules (and extensions), listed by author
- **clpa/ -** archive of the comp.lang.perl.announce Usenet newsgroup
- **doc/ -** Perl documentation and FAQs [getting a facelift/update soon]
- **indices/ -** various index views of CPAN
- **misc/** - the CPAN FAQ, the JAPH file, larry quotes, xearth coordinates for the mirrors, and other fun stuff.
- **modules/ -** modules (and extensions), listed by-authors/, by-module/ and by-category/
- **ports/** non-UNIX Perl ports, both as executables and source
- **scripts/ -** Perl utilities that aren't modules.
- **src/ -** The Perl source distribution.

Getting to know modules/

- **00modlist.long.html** - Long annotated list of modules.
- **01modules.index.html** - All distributions currently on CPAN alphabetically sorted by namespace and filename.
- **01modules.mtime.html** - the 150 most recent uploads ordered by timestamp.
- **02packages.details.txt.gz** - Long list of module, version, distribution mainly used by CPAN.pm
- **03modlist.data.gz** - Machine readable file used by CPAN.pm containing module name, dslip information, author id and category.
- **04pause.html** - THE document to read if you are wishing to be a CPAN author.
- **INSTALL.html** - What to do once you have downloaded a module and want to install it.
- **by-authors/** - hard to navigate unless you know the module and author you are looking for.
- **by-category/** - modules sorted by category as listed in the 00modlist.
- **by-module/** - modules listed by namespace, e.g. Foo::Bar would be located in by-module/Foo/
- **dbperl/** - vestige of the wild days of Perl4

Finding what you want.

1. Get some definition of the problem you are trying to solve to narrow your search area.
2. Browse through http://www.cpan.org/modules/00modlist.long.html as the modules are organised by category. It is not comprehensive, but it may help give you a better idea of how modules are classified and organised.
3. Use a search engine. CPAN has several of them and, while each of them have their strengths, they are invaluable tools for navigating CPAN.

Finding Obsolete/Older Modules

Sometimes you may find a need for an older version of a module and find that it has been deleted from the authors directory on CPAN. There are two ways you can possibly find what you are looking for:

- Write the author and ask for a copy if they still happen to have it kicking around (The Hard Way).

- Andreas Koenig has made BackPAN available which has most or all of every module ever submitted to PAUSE in one convenient archive in the familiar by-author/ directory(The Easy Way).

Search engines: search.cpan.org

1. Created by Graham Barr in early 1999
2. It’s strengths are a nice, easy to read GUI, the integrated CPAN testers information for each module and HTML documentation for the module.
3. You can search by module, distribution, author and documentation. A distribution contains one or more modules.
4. It’s busy. Best time to use it is after 1pm CST and before 7am CST.
5. Read the FAQ http://search.cpan.org/SearchFAQ/CPANSearchFAQ.html

Search engines: kobesearch.cpan.org

1. Like search.cpan, it has a search engine that searches through distributions, modules, authors and documentation.
2. It has more sophisticated search query capabilities.
3. It has a module "quickinstall" feature you can click on via your browser and it does the rest with the help of CPAN.pm.
4. It has the complete and current perl docs in HTML at http://perldoc.cpan.org/

Search engines: WAIT and perldoc.com

1. Best places to go if you are looking for perl core or module documentation with search capabilities.
2. perldoc.com for perl core documentation
3. WAIT for module documentation on the PAUSE.

III. Managing

Now What?

So you found the module of your dreams and you want to know if it’s any good or not. I personally use a rather simple algorithm:

- Does it have documentation and, if so, is it useful?
- Is the author known to have worthwhile modules/code?
- Does the code in the module look clean and as though the author put some thought into it.
- Are there any tests for the module? [i.e. does ‘make test’ run any tests of the module]
- Is the module discussed by any other Perl users as being good, bad or ugly?
- Does it work as advertised?
- If none of the above are true, does the author respond to helpful feedback?

What modules are already installed?

- `perldoc perllocal`
- `C:\>ppm query` for ActivePerl
- use the ExtUtils::Installed module

```
use ExtUtils::Installed;
my $instmod = ExtUtils::Installed->new()
 foreach my $module ($instmod->modules()) {
  my $ver = $instmod->version($module) || “???”;
  print “$module -- $version\n”;
}
```

- Use Tom Christiansen’s pmtools suite at

http://language.perl.com/misc/pmtools-1.00.tar.gz

Installing Modules

1. The easy way is to use CPAN.pm
   - `perl -MCPAN -e ‘install Foo::Bar’`

2. Manually
   1. unzip the distribution
   2. `perl Makefile.pl`
   3. `make`
   4. `make test`
   5. `make install`

   Read the perlmodinstall.pod documentation for more detailed information on manually installing modules.

Installing modules in batches

- Use bundles with CPAN.pm

A bundle is a module in the Bundle:: namespace that looks like a module but is really a list of modules for CPAN.pm to install.

```
cpan> b
```

will list available bundles on CPAN and

```
cpan> b Bundle::DBI
```

will give extended information about the bundle.

```
 cpan> install Bundle::DBI
```

will install Bundle::DBI

CPAN.pm-fu

- Use ‘look’ to make changes to a module before installing it

  ```
  cpan> look Date::Manip
  ```

  - This will download the module into the .cpan working directory, uncompress it and drop you into a subshell where you can edit what you need, exit the shell and then tell the CPAN.pm to install your edited version of the module.

- Use a local CPAN CD-ROM instead of an http/ftp archive. The urllist configuration parameter will accept a filename in the form of a URL.

  ```
  cpan>o conf urllist push file://host/mountpoint/CPAN
  cpan>o conf commit
  ```

- Tune your own CPAN.pm MyConfig.pm. This file should live in $HOME/.cpan/CPAN which will override the system configuration. This is useful for testing or if you think your Sysadmin is a lazy slacker who didn’t configure the system-wide CPAN/Config.pm properly. :-)

CPAN.pm-fu II

- Use readme to browse and assess modules before installing them.

```
cpan>readme Date::Manip
```

- Use CPAN::Site and its utilities to make your own ‘pseudo’ CPAN for modules you cannot or do not wish to upload to PAUSE.

Generate your local CPAN on an anonymous FTP, HTTP or NFS server: Then configure CPAN.pm to point the urllist at your local CPAN and then use

```
  perl -MCPAN::Site -e shell
```

to install your local modules.

Autobundle

Autobundle will, if no arguments are given, inventory all modules installed on the system and make a system-wide bundle which has numerous uses such as

•keeping a cluster of modules current
•distributing modules to a number of machines with consistency
•distributing a ‘favorite modules’ bundle to others.

```
cpan> autobundle Date::Christmas Date::Manip
Package namespace         installed    latest  in CPAN file
Date::Christmas                1.01      1.02  H/HF/HFB/Date-Christmas-1.02.tar.gz
Date::Manip                    5.39      5.39  S/SB/SBECK/DateManip-5.39.tar.gz

Wrote bundle file
    /home/chaos/monkey/.cpan/Bundle/Snapshot_2001_07_09_01.pm
```

The "Snapshot_2001_07_09_01.pm" file may then be used with the CPAN.pm

```
perl -MCPAN -e 'install Bundle::Snapshot_2001_07_09_01'
```

Module Testers

•Graham Barr and Chris Nandor started what is known as cpan-testers to collect reports on modules by platform and OS for modules on the cpan. This information is available at http://testers.cpan.org/ or with any module on search.cpan.org. If you would like to get more involved and help in the ongoing mission, join the mailing list at http://lists.perl.org/showlist.cgi?name=cpan-testers

•Michael Schwern is working on a project called CPANTS, the CPAN Testing Service which will provide feedback for module authors and, hopefully, raise the awareness and quality of the modules on CPAN. The mailing list is at http://lists.perl.org/showlist.cgi?name=perl-qa

•Give constructive feedback to authors if you find problems and positive feedback if you like the module or find new uses.

IV. Contributing

Register with the PAUSE

Before you can upload your modules to CPAN you must first get a PAUSE/CPAN user ID by sending an email to modules@perl.org with the following:

- your name
- email address
- URL of your homepage
- your chosen ID between 4 and 9 characters in length. Check on CPAN first to make sure it is not already taken.
- a brief description of what you are planning on submitting

Be prepared to wait for a response as all that can be automated, has been so a human being must do much of this work. Once you have received your ID but haven’t had any comments regarding your submission do not be discouraged or alarmed and proceed with uploading your work.

Register your namespace

- CPAN depends on authors respecting each others namespace
- Send an email to modules@perl.org to enquire about a name that you are wishing to use as there may be a better name for it or it may clash with an existing name.
- Be prepared to wait
- Be prepared to be flexible
- Respect the advice of the modules group as they have all been doing this for a long time and have a much broader perspective on the issues involved with running such an enormous archive.

Use h2xs

So, you have an idea or a module that you want to contribute to CPAN and want to know where to begin.

- Use h2xs
- Use h2xs
- Did I mention to use h2xs?

Originally created as a tool to convert C header files into XS modules, h2xs is your friend in making modules since it takes much of the tedious tasks out and lets you get right down to work.

```
h2xs -AX -n Foo::Bar
```

will create the skeleton for the module as well as the Makefile.PL.

An ExtUtils::MakeModule utility is about to be released and may well replace h2xs in it’s module skeleton creating capacity.

PODs for Everyone

h2xs creates a minimal POD section for authors so USE IT! :)

- You don’t need to write a book but a few lines about the module and what it does has not been proven to be harmful to your health.
- Include a brief description in the ‘description’ section to make it easier on the indexers and the search engines to display this very important bit of information for the users.
- Include a small example of code showing how one might use your module
- Include a copyright/license statement

USE A VERSION NUMBER!

- CPAN.pm, MakeMaker and other utilities rely on version numbers. A version number is defined as being a floating point number with at least 2 digits after the decimal, e.g. 1.02 not 1.0.2.
- Users rely on version numbers
- One would think this is a simple detail rarely overlooked but the http://www.cpan.org/modules/02packages.details.txt.gz would indicate otherwise in the abundance of 'undef' versions.
- If you want to release a 'beta' version to CPAN but don't want CPAN.pm to install it as the latest version use an "_" followed by 2 digits in the version to distinguish it from the production version, e.g. 1.02_01

License your module

We don’t care what license you use but pick one and include it in your module. There are some people who like to place the license in the README or in a LICENSE file but we would like to also see it place in the module POD itself. ExtUtils::ModuleMaker will give you a choice of 21 or more different licenses to choose from.

- Select a license that best suits you
- Educate yourself in the different licenses and the issues they attempt to solve
- Consider your audience
- Don’t be Perry Mason :-)

make test

- Write a test or two for your module, just something that will indicate that it works the way it should.

- Look at other modules with tests to get an idea of how to make a useful test for your module.

- The CPAN-testers will submit reports on you module only if you have included tests in your distribution.

MakeMaker

Once you have the module ready for upload to PAUSE, MakeMaker has a few features to assist in creating a distribution.

- `make distcheck` - this will compare the files in the build directory and the files listed in the MANIFEST and report any discrepancies.
- `make manifest` - this rewrites or creates the MANIFEST file including all the files found in the distribution
- `make disttest` - creates a distdir, a directory with `$(DISTNAME)-$VERSION`, runs a `perl Makefile.PL`, a `make` and `make test` in that directory.
- `make dist` - this will create a tar file of your distribution ready for uploading to the PAUSE. *PLEASE USE THIS*

In Absentia Module Maint

Sometimes an author goes missing and will be unresponsive to your email laden with feature requests, feedback or patches causing much frustration and consternation. CPAN doesn’t have a stated policy on this matter and mostly relies on the authors and users to work out the details. However, there are times and extenuating circumstances that may get the CPAN involved in the situation.

- Be courteous and considerate
- Make an earnest attempt to contact the author
- Be patient.
- Keep in mind that you are dealing with a person who invested time and care into something. A little respect goes a long way.

If you still believe that some intervention is required on the issue, e-mail cpan@perl.org and modules@perl.org with a detailed description of the module, the issue and what action you think is needed.

V. MIRRORING

Selecting a Mirror

If you decide you would like to have a copy of CPAN closer to home or are just trying to choose a good and stable mirror to use with CPAN.pm there are a few things to help you find what you need.

- Choose a mirror that is stable. CPAN has been blessed with very good mirrors and a very high retention of those mirrors. If we list it, it is likely to be around for a while.
- Look for good bandwidth and responsiveness. The CPAN mirror database has started including bandwidth size with the information when available.
- Check the status of the mirror on the CPAN mirror status page at http://mirror.cpan.org/status.cgi which checks every mirror 4 times daily for freshness.
- select an rsync archive when available

Mirroring CPAN

Mirroring CPAN is very simple. You'll need the following.

- Good connectivity
- 1GB+ of disk space
- An ftp or rsync client
- Use either mirror or rsync to make it happen.

If you don't wish to be a public mirror you do not need to contact CPAN for permission. Public mirrors may use the on-line form at http://mirror.cpan.org/mirror.cgi or use the template at the top of the MIRRORED.BY file on CPAN.

The CPAN Cluster

Ask Björn Hansen has created a replacement of the CPAN multiplexer that works with http and ftp sites through a bit of DNS magic.

http://www.cpan.org/ and ftp://ftp.cpan.org/ are a cluster of boxes that will attempt to serve you the box closest to you and will hopefully be faster as well.

Send email to cpan@perl.org if you have problems with any of these mirrors or if you would like to know more and get involved.

Resources

- http://perldoc.cpan.org/ - read perlmod, perlmodlib, perlnewmod, ExtUtils::MakeMaker and CPAN.pm
- http://www.cpan.org/misc/cpan-faq.html - Read it as it contains a lot more information than you think.
- http://lists.cpan.org/search.cgi?cat=CPAN - for all the mailing lists related to CPAN
- http://pause.cpan.org/ - PAUSE
- http://testers.cpan.org/ - the CPAN testers results page
- http://search.cpan.org/ , http://kobesearch.cpan.org/ , http://wait.cpan.org/ - search engines
- http://mirrors.valueclick.com/backup.pause/ - BackPAN archive of old modules
- http://www.cpan.org/ports/index.html - a list of all the known ports listed by platform
- http://mirror.cpan.org/ - for all your CPAN mirror needs.
- http://bookmarks.cpan.org/ - for a collection of useful URLs