Many modules, three Perls and one UNIX

Anton Berezin
tobez@tobez.org

Overview

- FreeBSD ports collection
- CPAN vs. ports collection
- Perldelta 5.005_03/5.6.1/5.8.0
- Problems & solutions
- Future directions

Overview

- FreeBSD ports collection
- CPAN vs. ports collection
- Perldelta 5.005_03/5.6.1/5.8.0
- Problems & solutions
- Future directions

FreeBSD ports collection

The FreeBSD Ports and Packages Collection offers a simple way for users and administrators to install

applications.

There are currently around 8,500 individual ports in the ports collection. 956 of those are perl modules.

FreeBSD ports collection

In most cases, one just does something like this:

```
# cd /usr/ports/devel/p5-Locale-Maketext-Lexicon
# make install clean
```

This takes care of dependencies, too.

FreeBSD ports collection

Under the hood, every port is a collection of a Makefile, distribution info, package description, and a packing list.

A simple port Makefile:

```
PORTNAME=       Text-Balanced
PORTVERSION=    1.91
CATEGORIES=     textproc perl5
MASTER_SITES=   ${MASTER_SITE_PERL_CPAN}
MASTER_SITE_SUBDIR=     ../../authors/id/D/DC/DCONWAY
PKGNAMEPREFIX=  p5-

MAINTAINER=     tobez@FreeBSD.org
COMMENT=        Text::Balanced - extract delimited text sequences from strings

PERL_CONFIGURE= yes

MAN3=           Text::Balanced.3

.include <bsd.port.mk>
```

FreeBSD ports collection

Under the hood, every port is a collection of a Makefile, distribution info, package description, and a packing list.

A simple port Makefile:

```
PORTNAME=       Text-Balanced
PORTVERSION=    1.91
CATEGORIES=     textproc perl5
MASTER_SITES=   ${MASTER_SITE_PERL_CPAN}
MASTER_SITE_SUBDIR=     ../../authors/id/D/DC/DCONWAY
PKGNAMEPREFIX=  p5-

MAINTAINER=     tobez@FreeBSD.org
COMMENT=        Text::Balanced - extract delimited text sequences from strings

PERL_CONFIGURE= yes

MAN3=           Text::Balanced.3

.include <bsd.port.mk>      <---- the Evil piece
```

FreeBSD ports collection

```
.include <bsd.port.mk>
```

```
$ cd /usr/port/Mk
$ ls -l bsd.port.mk
-rw-r--r--  1 root  wheel  139048 Apr 22 15:10 bsd.port.mk
```

Overview

- FreeBSD ports collection
- CPAN vs. ports collection
- Perldelta 5.005_03/5.6.1/5.8.0
- Problems & solutions
- Future directions

CPAN vs. ports collection

We all love and use CPAN. We know it is nice. We know it is easy.

And it has more than puny 956 modules, too!

And

```
# perl -MCPAN -e 'install "Locale::Maketext::Lexicon"'
```

is shorter than

```
# cd /usr/ports/devel/p5-Locale-Maketext-Lexicon
# make install clean
```

And it takes care of the dependencies, too!

CPAN vs. ports collection

Problem: CPAN.pm, or at least its older versions, oftentimes insist on installing the latest and greatest Perl, which is usually not what you want.

Solution: There is no good solution. You can try and install a newer `Bundle::CPAN`, but then it will helpfully proceed installing new Perl as a dependency anyway.

Don’t hit me much if this has been fixed.

CPAN vs. ports collection

Problem: Deinstallation

`ExtUtils::MakeMaker` generates a Makefile with the following:

```
UNINSTALL =   $(PERL) -MExtUtils::Install \
-e 'uninstall($$ARGV[0],1,1); print "\nUninstall is deprecated. Please check the";' \
-e 'print " packlist above carefully.\n  There may be errors. Remove the";' \
-e 'print " appropriate files manually.\n  Sorry for the inconveniences.\n"'
```

Solution: FreeBSD has `BSDPAN`

BSDPAN

BSDPAN is a small collection of Perl modules located at `APPLLIB_EXP`, a portion of `@INC` which in general precedes `PRIVLIB`.

Thus, it has a chance to override certain subs in `ExtUtils::MM_Unix` and in `ExtUtils::Packlist`.

Meaning that even modules installed directly from CPAN get an entry in the FreeBSD database of installed packages.

CPAN vs. ports collection

Problem: Overwritten files
The INSTALLDIRS => 'perl' modules do not play well with the concept of knowing what and where your files are, especially with Perl present in the base system. FreeBSD 4.X can retaliate during make installworld, too.

Solution: SITELIB comes before PRIVLIB in all FreeBSD Perls.

Yeah, but what about INSTALLDIRS => 'perl' modules?

Don't worry, BSDPAN won't let you.

CPAN vs. ports collection

Last, but not least: using ports collection preserves POLA for FreeBSD users.

POLA = Principle Of Least Astonishment

It is convenient to be able to install **all** software via a single mechanism.

It is also convenient to be able to get the information about all installed software in a uniform manner.

(but see BSDPAN for the counterpoint to the last sentence)

Overview

- FreeBSD ports collection
- CPAN vs. ports collection
- Perldelta 5.005_03/5.6.1/5.8.0
- Problems & solutions
- Future directions

Three Perls

- 5.005_03 is still in the base system in FreeBSD 4.X
- Perl 5.6.1 is available as ports/lang/perl5
- Perl 5.8.0 is available as ports/lang/perl5.8
- FreeBSD 5.X does not have Perl in the base system
- though lang/perl5 is installed by default there

Delta 5.6.1 - 5.005_03

(in brief)

- Lots of fixes
- Features:
  - our
  - use warnings
  - @+ and @-
- New modules in the core
- New bugs

Delta 5.6.1 - 5.005_03

```
eval { my $x = $^S; };
print "\$^S is $^S\n";
__END__
```

This bug is **only** present in 5.6.1. 5.005_03, 5.6.0, 5.8.0 do not have it.

Fixed in forthcoming (?) 5.6.2

Fixed in FreeBSD port

Delta 5.8.0 - 5.6.1

(in brief)

- Lots of fixes
- Features:
  - working ithreads
  - reliable signals
  - attributes
  - real unicode
- An amazing number of new modules in the core
- Sometimes misfeatures:
  - reliable signals
- New bugs

Delta 5.8.0 - 5.6.1

```
$ perl -MConfig -e 'use lib $Config{installsitelib}'
```

Fixed in forthcoming 5.8.1

Fixed in FreeBSD port

Delta 5.8.0 - 5.6.1

```
use IO::File;

my $null = IO::File->new("/dev/null", "r+");
die "open(\"/dev/null\"): $!\n" unless defined $null;
open(STDIN, "<& " . fileno($null)) or die("dup STDIN: $!\n");
```

Fixed in forthcoming 5.8.1

NOT fixed in FreeBSD port

Delta 5.8.0 - 5.6.1

- reliable signals are sometimes too reliable
- there were some examples during the threads talk
- one more example: FastCGI:
  - SIGTERM is handled in an application
  - the code is modified
  - the application is killed to do a refresh
  - nothing happens
  - the next request arrives
  - ***BOOM***
  - logs say: unexpected exit with code 0 condition

Overview

- FreeBSD ports collection
- CPAN vs. ports collection
- Perldelta 5.005_03/5.6.1/5.8.0
- Problems & solutions
- Future directions

Perldelta-related problems

- modules which really need 5.8.0
- modules which really need 5.6.1
- modules which think they need 5.6.1
- modules which adapt
- modules which do not need much

Really need 5.8.0

- typically the ones that work with attributes
- not much we can do except telling the user
- examples in the ports collection:
  - none

Really need 5.6.1

- the ones which use weak references:
  - Object::Realize::Later
  - Mail::Box v2
- the ones which do XS with 5.6.1-specific things:
  - Coro
- not much we can do except telling the user

Modules which think they need 5.6.1

- these are the most fun
- typically those that just use ‘our’ and/or ‘use warnings’
- in most cases a simple substitution will do

Modules which think they need 5.6.1

```
${PERL} -pi -e 's/^our\s+([\$\@\%]\w+)/use vars qw($$1);$$1/;' \
            -e '$$_="" if /require 5/;' \
            -e '$$_="" if /use warnings/;' \
      ${WRKSRC}/Extra.pm
```

....or somesuch

Modules which think they need 5.6.1

The thorough testing is required, though. Consider:

```
my $x = 5;
{
    our $x = 6;
}
print "$x\n";
```

prints 5, while

```
my $x = 5;
{
    use vars qw($x); $x = 6;
}
print "$x\n";
```

prints 6.

Modules which adapt

- these work fine with 5.005_03
- but they can do more with 5.6:
  - Scalar-List-Utils
  - Regexp::Common
- nothing to worry about

Modules which do not need much

- no comments

Other cases

- modules which are more recent or the same in base 5.8.0:
  - Digest::MD5
- modules which are more recent or the same in base 5.6.1:
  - File::Spec
- modules which are specifically for 5.6.1:
  - Encode::compat
- not much we can do except telling the user
- special dependency handling:
  - those that depend on Digest::MD5
  - those that depend on File::Spec

Overview

- FreeBSD ports collection
- CPAN vs. ports collection
- Perldelta 5.005_03/5.6.1/5.8.0
- Problems & solutions
- Future directions

Future directions

- add more nice modules
- retire 4.x branch
  - and with it, finally, 5.005_03
- make lang/perl* more robust
  - meaning: don't let Configure guess anything :-(
- make different Perls peacefully coexist on the same system
  - preferably also same Perls built differently (threads etc.)
  - including the modules
  - without loosing bits and pieces of the package database
  - yeah,
  - pkg_info p5-581-ithr-Very-Long-Module-Name
  - pkg_info p5-561-pmalloc-Very-Long-Module-Name
  - right

Any questions?

Any questions?

References

- http://www.FreeBSD.org/
- http://www.FreeBSD.org/ports/
- http://theoryx5.uwinnipeg.ca/CPAN/cpan-search.html
- The FreeBSD Porter's Handbook

The online version of the slides will be in a few days here:

- http://www.tobez.org/perl/SPW2003/

Many modules, three Perls and one UNIX

Thank you!

Anton Berezin
tobez@tobez.org