Ruby on pkgsrc affair

Takahiro Kambe / 神戸 隆博

taca@back-street.net

taca@NetBSD.org / taca@contao.jp

@_taca_ / @_taca_en

About me

- One of a NetBSD developers since Aug. 2000
    - area: pkgsrc (packaging system)
        - Ruby, PHP and more
- A manager of Japn Contao User Group
    - PHP based Web CMS
- Translation of a few Open Source software
- An employee of a small company (in real life)
    - Not venture nor start up
    - 主に学校向けのPC/iPad導入や保守の支援、何故か小さなプログラム開発、Web担当

NetBSD

- BSD derived Operating System

  - http://www.NetBSD.org

  - from http://www.slideshare.net/junebihara18/netbsd-on-the-road-2016

    - Start from 21 March 1993

    - Supporting platforms (ports): 58

    - Supporting CPU architectures: 15

  - "Write once, Run everywhere, Java is optional."

  - Brothers

    - FreeBSD, OpenBSD, DragonFlyBSDなど

pkgsrc

- Framework for building packages
  - http://www.pkgsrc.org
    - Originaly derived from FreeBSD's ports
- Supports multi platform: 17
  - "Write once, build, pakcage and run anywhere!"

```
% ls -F /usr/pkgsrc/mk/platform
AIX.mk          Darwin.mk       HPUX.mk       Minix.mk      QNX.mk
BSDOS.mk        DragonFly.mk    Haiku.mk      MirBSD.mk     README
Bitrig.mk       FreeBSD.mk      IRIX.mk       NetBSD.mk     SCO_SV.mk
CVS/            FreeMiNT.mk     Interix.mk    OSF1.mk       SunOS.mk
Cygwin.mk       GNUkFreeBSD.mk  Linux.mk      OpenBSD.mk    UnixWare.mk
```

Powered by
NetBSD®
www.NetBSD.org

pkgsrc

What pkgsrc will do?

- Packaging phase

    1. fetch

    2. extract

    3. patch

    4. configure

    5. build

    6. install (staged install)

    7. package

        - "a binary package" will be produced.

        - binary package name: (pkgbase)-(version).tgz

How it work?

- Some basic commands
  - bmake (BSD make derived from pmake)
  - pkg_install family commands
    - pkg_create, pkg_info, pkg_add
- Macros for bmake
  - Too many and complex: /usr/pkgsrc/mk
- A few supporting scripts
- bootstrap for none NetBSD platform

Using binary packages

- Commands of pkg_install family: pkg_add

  # pkg_add ruby23-base-2.3.1.tgz

- Install from CDN

  # PKG_PATH="http://cdn.NetBSD.org/pub/pkgsrc/packages/$(uname -s)/$(uname -m)/$(uname -r|cut -f '1 2' -d.)/All/"

  # export PKG_PATH

  # pkg_add ruby23-base-2.3.1.tgz

- Other commands:

  pkg_delete, pkg_info, pkg_admin

Past Ruby on pkgsrc

- 9 July 1999: Initial Ruby package
  - Ruby-1.3.4 (ruby-1.3.4-990625)
- 30 June 2001: First my concern
  - Ruby 1.6.4 (or Ruby 1.4.6)
    - ruby-base
    - ruby-gdbm, ruby-md5, ruby-readline, ruby-tcltklib, ruby-tk

Current Ruby packages

- Mutiple versions of Ruby support

  - Install at the same time.

  - ruby23, ruby22, ruby21 and ruby18

    - ruby-gdbm, ruby-readline, ruby-tk
      ruby-curses (ruby 18 only)
      ruby-fiddle (ruby21 and later)
      ruby-tk

  - Binary packages have prefix: "ruby23-"

Number of Ruby packages

- Rough counting:

  - 576

    - on my working space

    - whole pkgsrc contains more than 17500 pacakges

- Several developers add/update them.

Rubygem support

- Invented by other developer
    - Patch rubygems library of Ruby
    - Install temporary own root directory
    - Remove intermediate files: *.o
    - Create binary package.
- Installing multiple versions of gems are not supported.

Rails support

- Multiple versions of rails support.

  - ruby21-activesupport31

  - ruby21-activesupport32

- But, currently Rails 3.2 is only supported.

  - Simply man power problem due to:

    - Many depending packages

    - Is it really work enough?

How pkgsrc handles gems as packages friendly?

- Many gems

  - Sometimes diffrent version is required.

  - Not well versioned case.

  - Brute force solution

    - OVERRIDE_GEMSPEC

      - modify content of gemspec.

    - please refer redmine package:

http://cvsweb.netbsd.org/bsdweb.cgi/pkgsrc/devel/ruby-redmine/Makefile

How to contribute?

- Use it!

- Send bug reports.

- Create new pkgsrc.

- Request new pkgsrc.

  - http://www.netbsd.org/support/send-pr.html

- pkgsrc-wip project

  - https://pkgsrc.org/wip/

  - Use git as repository

Thank you!
おおきに、ありがとう。