SmartOS

pkgsrc on SmartOS

Jonathan Perkin
Software Engineer
Joyent

@jperkin
jperkin@joyent.com
jperkin@pkgsrc.org

Agenda

SmartOS

- Introduction

- History and background to SmartOS

- pkgsrc usage in SmartOS

- pkgsrc features we've developed

- pkgsrc development at Joyent

- Our pkgsrc wishlist

$ whoami

- Started working on pkgsrc in 2001

- Used pkgsrc on Solaris internally at BBC

- Worked on build systems and release engineering for MySQL/Sun/Oracle

- Now work full-time on pkgsrc for Joyent

```
$ uname -a
SunOS www0-rth.thls.bbc.co.uk 5.7 Generic sun4m sparc SUNW,SPARCstation-20

$ uptime
 12:28pm  up 5000 day(s),  2 users,  load average: 0.91, 0.89, 0.87
```

What is SmartOS - History

- Solaris 10 released by Sun in 2005

  - ZFS, DTrace, Zones, SMF

  - Open Source

- OpenSolaris continued the development towards Solaris 11

  - IPS, Crossbow, Laptop/Desktop support, GNU environment

- However, Oracle made Solaris closed source and proprietary in 2010

- illumos was born to continue development of the last available OpenSolaris code

  - Replace closed-source parts needed to operate the system

  - Continue feature development, working with other communities

What is SmartOS - illumos Distributions

- Diaspora of Solaris engineers from Oracle and community talent has to illumos distributions focused on specific areas of interest

  - Delphix continue DTrace and ZFS development for database virtualisation products

  - Nexenta and DEY focus on storage appliances and services

  - OmniTI produce OmniOS, a general-purpose illumos distribution for servers

  - OpenIndiana continues development of a desktop-oriented environment

  - Joyent produce SmartOS with a specific focus on cloud computing

What is SmartOS - Joyent

SmartOS

- Joyent operate a *high performance* public cloud. The key features of illumos we require which differentiate it are:

  - ZFS. Fast, scalable, robust storage

  - DTrace. Observability and debugging across *everything*

  - Zones. Low latency and highly customisable virtualisation

  - KVM. Ability to run customer's preferred OS (though we always recommend Zones wherever possible)

What is SmartOS - Thin Hypervisor

SmartOS

- SmartOS is a USB/network-booted thin hypervisor

    - Open source, available on GitHub

    - OS resides on USB stick or PXE

    - All disks are dedicated to zones

- The "Global Zone" is a ramdisk for zone administration, nothing else

    - `/usr` is a read-only lofs mount, changes to `/etc`, `/root`, etc. not permanent

    - Upgrades are trivial, replace `platform` directory and reboot

    - Latest "First Customer Ship" build available every 2 weeks

- Clear separation between OS (platform) and packages (pkgsrc)

    - `PREFER_PKGSRC=yes` (except for `solaris-pam`, `terminfo`)

    - Zone images are fully independent of OS version

- All work is done inside Zones

    - Zones are created from images, made up of a file system image (ZFS snapshot) and metadata (JSON)

    - `imgadm` and `vmadm` tools for management of images and virtual machines

    - OS zones are "sparse" and mount `/usr` from GZ

- KVM systems are just images too, and run `qemu` process inside a thin zone

    - Gain hosting features (throttling, resource management, etc.)

    - Additional security

Demo Time!

pkgsrc on SmartOS - Basics

SmartOS

- We track quarterly branches for each image version

  - `2012Q2 => base*-1.8.x, 2012Q4 => base*-1.9.x,` etc

- Separate images for 32-bit (`ABI=32, base`) and 64-bit (`ABI=64, base64`)

- Self-contained layout

  - `PREFIX=/opt/local`

  - `PKG_SYSCONFDIR=/opt/local/etc`

  - `PKG_DBDIR=/opt/local/pkg`

- Various package sets available

  - `base` for minimal install

  - `standard` for base + common utilities

  - `percona, riak` etc for specific purpose

pkgsrc development

- Our users demand additional pkgsrc features that we have developed:

  - Multi-architecture binaries and libraries

  - SMF support

  - "SmartOS is Not GNU/Linux" distribution (`PREFIX=/usr`)

pkgsrc development - multiarch

- Separate 32-bit and 64-bit images are a pain:

    - Confusing for users and customers who think it refers to the kernel architecture

    - Additional build and storage requirements

    - Solaris has had multiarch support since Solaris 7 in 1998, old-time users have come to expect it

pkgsrc development - multiarch #2

SmartOS

- High-level overview:

  - For multiarch-enabled packages, perform each build stage (extract, patch, ...) twice, once with `ABI=32`, once with `ABI=64`

  - `--bindir=` and `--libdir=` set based on `ABI`

  - Install both to the same `DESTDIR`, default `ABI` installed last

  - `PLIST` expansion to cover multiple entries

  - `isaexec` wrapper installed to handle binaries

  - Create package

- Enable with `MULTIARCH=yes` in `mk.conf`

pkgsrc development - multiarch #3

SmartOS

- Infrastructure (`mk/`) changes:

  - Each primary phase (e.g. '`do-extract`') has a wrapper (e.g. '`do-extract-multi`') which calls the primary target for each ABI

  - `mk/plist/plist-multiarch.awk` handles PLIST expansion

  - OPSYS-specific `{BIN,LIB}ARCHSUFFIX.{32,64}` variables

    - `BINARCHSUFFIX.32= /i86`

    - `BINARCHSUFFIX.64= /amd64`

  - `pkgtools/isaexec` package created, supports ABI environment variable

  - `pkg_install` supports new `@link` directive for `isaexec` hardlinks

pkgsrc development - multiarch #4

- Converting a simple autoconf/cmake package requires:

  - `USE_MULTIARCH= bin lib`

- Converting more complicated custom packages involves:

  - `USE_MULTIARCH= bin lib`

  - Change hardcoded `lib` to `lib${LIBARCHSUFFIX}`

  - Change hardcoded `bin` to `bin${BINARCHSUFFIX}`

  - Use `MULTIARCH_DIRS.{bin,lib}`

  - Use `MULTIARCH_SKIP_DIRS.{bin,lib}`

  - Complicated `WRKSRC` handling

- Developers, please stop writing your own build systems!

pkgsrc development - multiarch #5

SmartOS

- Current status

  - Large number of packages converted (`p5-*` automatically handled)

  - SmartOS "`trunk`" image uses multiarch by default, ~8,000 packages

  - `504 files changed, 3525 insertions(+), 902 deletions(-)`

- Developed in Joyent pkgsrc repository

  - `joyent/feature/multiarch/trunk -> joyent/release/trunk`

  - `joyent/feature/multiarch/YYYYQQ -> joyent/release/YYYYQQ_multiarch`

pkgsrc development - SMF

SmartOS

- Solaris 10 introduced the Service Management Framework as a replacement for the legacy `init.d` system. We need to support it.

  - `svccfg` to import a new manifest

  - `svcadm` to enable/disable/restart

  - `svcs` to display status

- Small number of infrastructure changes

  - New `mk/smf.mk`

  - `4 files changed, 131 insertions(+)`

- Manifest and method files current held externally similar to `LOCALPATCHES`

  - `${SMFBASE}/${PKGPATH}`

- Developed in `joyent/feature/smf/trunk` branch

pkgsrc development - SNGL

SmartOS

- "SmartOS is Not GNU/Linux", aka "snuggle"

  - Some users not keen on `/opt/local` layout, we can fix that!

  - Flexibility of zones allows `/usr -> /system/usr`, `/lib -> /system/lib`, etc.

  - pkgsrc built with `LOCALBASE=/usr PKG_SYSCONFDIR=/etc`

  - Symlinks in `/usr` for compatability, overridden by packages

- Works surprisingly well, with caveats

  - Most builtin checks are not `LOCALBASE=/usr` clean

  - Some packages (hi perl!) make assumptions about `/usr`

- Would be great to see other users of this, e.g. Linux

  - Developed in `joyent/feature/sngl/trunk` branch

pkgsrc development - gccruntime

- SmartOS does not provide GCC runtime (libgcc, libstdc++ etc), needs to come from pkgsrc

  - Split runtime out to `lang/gcc47-libs` package

  - Packages register full `DEPENDS` upon it, use GCC `-specs` override for `RPATH`

  - `USE_PKGSRC_GCC_RUNTIME=yes` to enable

  - Packages which `USE_LIBTOOL` handled automatically

- Packages which build their own shared libraries need special handling

  - `USE_GCC_RUNTIME=yes`

  - Not ideal, `compiler.mk` sourced early so needs to be set first

- Would like to integrate remaining `USE_GCC_RUNTIME` for 2013Q2

  - Currently developed in `joyent/feature/gccruntime/trunk`

pkgsrc development - misc

- OpenJDK7

    - Patches from SmartOS user "jesse_" to get latest OpenJDK working

    - Mostly working `openjdk7-current` package, would like to discuss *BSD status

- `_PBULK_MULTI` support for mysql* and percona

    - Need a good naming scheme, `p5-DBD-mysql51` / `p5-DBD-percona55`?

- Various small changes in `joyent/feature/miscfix/trunk`

pkgsrc development - git

- We develop everything with git / github

  - https://github.com/joyent/pkgsrc (thanks joerg!)

  - https://github.com/joyent/pkgsrc-wip

  - https://github.com/joyent/pkgsrc-joyent

- Cheap and fast branching makes it all possible

  - `upstream/trunk->trunk, upstream/pkgsrc_*->pkgsrc_* (pull --rebase)`

  - `trunk -> joyent/feature/*/trunk -> joyent/release/trunk`

  - `pkgsrc_YYYYQQ -> joyent/feature/*/YYYYQQ -> joyent/release/YYYYQQ`

  - `joyent/feature/*/YYYYQQ -> joyent/release/YYYYQQ_feature`

- Currently 5 active features (`gccruntime, miscfix, multiarch, smf, sngl`)

- Short-lived branches (`joyent/bulktest/*, pkggnudir, openssl1, etc`)

pkgsrc development - pbulk

- We build lots of binary packages

  - `/opt/pbulk` bootstrap + development tools

  - pbulk cluster of 6 client VMs (8 CPU, 8GB each), 1 master, + NFS

  - Can perform a full clean bulk build in under 2 days with `MAKE_JOBS=8`

  - Builds done inside chroot for concurrency

- Some changes required (in `joyent/feature/miscfix/trunk`)

  - Support creation/deletion of chroots + config file support

  - `SO_REUSEADDR` for `limited_list`

  - Enable rsync of `RESTRICTED` dependencies

  - Add `SPECIFIC_PKGS` support for full pkgsrc + selected pkgsrc-wip

pkgsrc development - Jenkins

- We use Jenkins for triggering bulk builds

    - Was in use by other parts of Joyent anyway

- GitHub plugin monitors for pushes in `joyent/release/*` branches

    - Build triggered for every push

    - Limited to one concurrent build per branch,

- Upstream builds triggered via Jenkins "cron"

    - Tracked directly from `jsonn/pkgsrc.git`

    - 4 builds per week (2 x 32-bit, 2 x 64-bit)

    - Results sent directly to pkgsrc-bulk@netbsd.org

- Provides quite a nice interface, plus Jabber notification etc.

pkgsrc wishlist

- Things we really want at some point:

    - Support for multiple repositories in pkgin

    - Decent upgrade support / config file management

    - Multi-packages (users require different `PKG_OPTIONS`)

    - Additional pbulk-multi support (mysql, postgres, etc.)

- Things which would be nice to have:

    - apt/dpkg backend support

    - Better `pkg_alternatives` support

    - pkgin package aliases (#1 confusion for users is `scmgit`, #2 is `gcc47`)

    - Working (and updated) xorg server

Thank You!

SmartOS

- "All this is possible thanks to the unique way the BBC is funded"

- Thank you for all your development work and pkgsrc commits

  - We couldn't handle the pace of upgrades ourselves

  - Your work is used further than perhaps you realise

- Let's keep going!

  - Regular binary builds for other platforms (OSX, Cygwin)

  - Become the de-facto package manager on more platforms

  - Users => Fixes => Developers => Users => ...

Thanks!

SmartOS

- www.joyent.com - high-performance VMs and private cloud infrastructure

    - SmartDataCentre (SDC) powered, SmartOS + additional tooling

    - REST APIs for managing VMs

    - Cloud Analytics (DTrace) plus other goodies

    - Use "FREEJOY" promo code for $125 free credit (2 months running a 1GB SmartMachine)

- www.smartos.org - open source OS for running VMs

    - Contribute your own images to http://datasets.at/ or publish yourself

- Questions?