Controlling Software Environments with GNU Guix

Ludovic Courtès

Inria Bordeaux Sud-Ouest
November 2016
The difficulty of keeping software environments under control.
#1. Upgrades are hard.
Distribution Upgrade of all the files:

WARNING

Following the upgrade instructions found in the release notes is the best way to ensure that your system upgrades from one major Debian release to another (e.g. from lenny to squeeze) without breakage!

These instructions will tell you to do a dist-upgrade (instead of upgrade) in the case of apt-get or full-upgrade (instead of safe-upgrade in the case of aptitude) at least once. So you would have to type something like

```
# aptitude full-upgrade
```
4.2. Check system status
   4.2.1. Review actions pending in package manager
   4.2.2. Disabling APT pinning
   4.2.3. Checking packages status
   4.2.4. The proposed-updates section
   4.2.5. Unofficial sources

4.3. Preparing sources for APT
   4.3.1. Adding APT Internet sources
   4.3.2. Adding APT sources for a local mirror
   4.3.3. Adding APT sources from optical media

4.4. Upgrading packages
   4.4.1. Recording the session
   4.4.2. Updating the package list
   4.4.3. Make sure you have sufficient space for the upgrade
   4.4.4. Minimal system upgrade
   4.4.5. Upgrading the system

4.5. Possible issues during upgrade
   4.5.1. Dist-upgrade fails with “Could not perform immediate configuration”
   4.5.2. Expected removals
   4.5.3. Conflicts or Pre-Depends loops
#2. Stateful system management is intractable.
$DISTRO

$DISTRO
$DISTRO$

- apt-get update
  - state 1$_a$

$DISTRO$

- apt-get update
  - state 1$_b$
$DISTRO

apt-get update

state 1\textsubscript{a}

apt-get install foo

state 2\textsubscript{a}

apt-get update

state 1\textsubscript{b}

apt-get remove bar

state 2\textsubscript{b}
$DISTRO

apt-get update

state 1

apt-get install foo

state 2

apt-get remove bar

state 3

apt-get update

state 1

apt-get remove bar

state 2

apt-get install foo

state 3
$DISTRO$

state 1
- apt-get update
- apt-get install foo

state 2
- apt-get remove bar

state 3
- apt-get remove bar

$DISTRO$

state 1
- apt-get update

state 2
- apt-get remove bar

state 3
- apt-get install foo

= ?
#3. Entropy keeps increasing.
Here is an example of loading a module on a Linux machine under bash.

% module load gcc/3.1.1
% which gcc
/usr/local/gcc/3.1.1/linux/bin/gcc

Now we'll switch to a different version of the module

% module switch gcc gcc/3.2.0
% which gcc
/usr/local/gcc/3.2.0/linux/bin/gcc
Application-level package managers

- **Anaconda** - a package manager for Python
- **Assembly** - a partially compiled code library for use in Common Language Infrastructure (CLI) deployment, versioning and security.
- **Biicode** - a file-focused dependency manager for C/C++ languages and platforms (PC, Raspberry Pi, Arduino).
- **Bower** - a package manager for the web.
- **UPT** - a fork of Bower that aims to be a universal package manager, for multiple environments and unlimited kind of package.
- **Cabal** - a programming library and package manager for Haskell
- **Cargo** - a package manager for Rust (programming language)
- **CocoaPods** - Dependency Manager for Objective-C and RubyMotion projects
- **Composer** - Dependency Manager for PHP
- **CPAN** - a programming library and package manager for Perl
- **CRAN** - a programming library and package manager for R
- **CTAN** - a package manager for TeX
- **DUB** - a package manager for D
As of npm@2.6.1, the npm update will only inspect top-level packages. Prior versions of npm would also recursively inspect all dependencies. To get the old behavior, use npm --depth 9999 update, but be warned that simultaneous asynchronous update of all packages, including npm itself and packages that npm depends on, often causes problems up to and including the uninstallation of npm itself.

To restore a missing npm, use the command:

```
curl -L https://npmjs.com/install.sh | sh
```
#!/bin/bash

pip install "$1" &
easy_install "$1" &
brew install "$1" &
npn install "$1" &
yum install "$1" & dnf install "$1" &
docker run "$1" &
pkg install "$1" &
apt-get install "$1" &
sudo apt-get install "$1" &
steamcmd +app_update "$1" validate &
git clone https://github.com/"$1"/"$1" &
cd "$1";./configure;make;make install &
curl "$1" | bash &
Giving up?
Giving up?

→ “app bundles” (Docker images & co.)
“Debian and other distributions are going to be that thing you run docker on, little more.”

— Jos Poortvliet, ownCloud developer
It’s also that thing you run inside Docker!
ADD file:e5a3d20748c5d3dd5fa11542dfa4ef8b72a0bb78ce09f6dae30eff5d045c67aa in /

125 mb

CMD "/bin/bash"

0 bytes

RUN apt-get update && apt-get install -y --no-install-recommends ca-certificates curl wget && rm -rf /var/lib/apt/lists/*

44 mb

RUN apt-get update && apt-get install -y --no-install-recommends bazaar git mercurial openssh-client subversion procps && rm -rf /var/lib/apt/lists/*

123 mb
Over 30% of Official Images in Docker Hub Contain High Priority Security Vulnerabilities

Docker Hub is a central repository for Docker developers to pull and push container images. We performed a detailed study on Docker Hub images to understand how vulnerable they are to security threats. Surprisingly, we found that more than 30% of images in official repositories are highly susceptible to a variety of security attacks (e.g., Shellshock, Heartbleed, Poodle, etc.). For general images – images pushed by docker users, but not explicitly verified by any authority – this number jumps up to ~40% with a sampling error bound of 3%.
October 20, 2016

Container App ‘Singularity’ Eases Scientific Computing

Tiffany Trader

HPC container platform Singularity is just six months out from its 1.0 release but already is making inroads across the HPC research landscape. It’s in use at Lawrence Berkeley National Laboratory (LBNL), where Singularity founder Gregory Kurtzer has worked in the High Performance Computing Services (HPCS) group for 16 years, and it’s going into other leading HPC centers, including the Texas Advanced Computing Center (TACC), the San Diego Supercomputing Center (SDSC) and many more sites, large and small.

https://www.hpcwire.com/2016/10/20/singularity-containers-easing-scientific-computing
Adios apt and yum? Ubuntu’s snap apps are coming to distros everywhere

More secure replacement for debs coming to Fedora, Arch, Debian, and more.

JON BRODKIN  -  6/14/2016, 7:00 PM
THE FUTURE OF APPLICATION DISTRIBUTION

The days of chasing multiple Linux distributions are over. Standalone apps for Linux are here!

FIND OUT HOW
“app bundles” are headed wrong

- difficulty to **compose** software packages
- wrong **abstraction level**: image vs. package
- **hardly reproducible**: we have the bits, not the source
- makes it hard to **customize & experiment**
Make packaging great again!
Guix

1. transactional package manager
2. software environment manager
3. APIs & tools to customize environments
4. packaging tools
$ guix package -i gcc-toolchain coreutils sed grep
...

$ eval 'guix package --search-paths'
...

$ guix package --manifest=my-software.scm
...

Want to get started hacking on hwloc?
Want to get started hacking on hwloc?

A simple matter of installing the deps, right?
$ guix environment --container hwloc
...

$ guix environment --container hwloc \\n   --ad-hoc git autoconf automake gdb
...

$ guix build hello

isolated build: chroot, separate name spaces, etc.
$ guix build hello
/gnu/store/ h2g4sf72... -hwloc-1.11.2

hash of all the dependencies
$ guix build hello
/gnu/store/ h2g4sf72... -hwloc-1.11.2

$ guix gc --references /gnu/store/...-hwloc-1.11.2
/gnu/store/...-glibc-2.24
/gnu/store/...-gcc-4.9.3-lib
/gnu/store/...-hwloc-1.11.2
$ guix build hello
/gnu/store/ h2g4sf72... -hwloc-1.11.2

$ guix gc --references /gnu/store/...-hwloc-1.11.2
 /gnu/store/...-glibc-2.24
 /gnu/store/...-gcc-4.9.3-lib
 /gnu/store/...-hwloc-1.11.2

(nearly) bit-identical for everyone
Can we go beyond mere reproducibility and support experimentation?
Reproducible and User-Controlled Software Environments in HPC with Guix

Ludovic Courtès¹ and Ricardo Wurmus²

¹ Inria, Bordeaux, France
² Max Delbrück Center for Molecular Medicine, Berlin, Germany

Abstract. Support teams of high-performance computing (HPC) systems often find themselves between a rock and a hard place: on one hand, they understandably administrate these large systems in a conservative way, but on the other hand, they try to satisfy their users by deploying up-to-date tool chains as well as libraries and scientific software. HPC system users often have no guarantee that they will be able to reproduce results at a later point in time, even on the same system—software may have been upgraded, removed, or recompiled under their feet, and they have little hope of being able to reproduce the same software environment elsewhere. We present GNU Guix and the functional package management paradigm and show how it can improve reproducibility and sharing among researchers with representative use cases.
Creating package variants at the command line
$ guix build hwloc \
   --with-source=./hwloc-42.0rc1.tar.gz
...

...
$ guix build hwloc \
   --with-source=./hwloc-42.0rc1.tar.gz
...

$ guix package -i mumps \
   --with-input=scotch=pt-scotch
...
Your personal packages or variants in **GUIX_PACKAGE_PATH**!
GuixSD: declarative OS config

(timezone "Europe/Paris")
(locale "en_US.utf8")

(bootloader (grub-configuration
    (device "/dev/sda")))

(mapped-devices (list (mapped-device
    (source "/dev/sda3")
    (target "home")
    (type luks-device mapping)))

(file-systems (cons* (file-system
    (device "root")
    (title 'label)
    (mount-point "/")
    (type "ext3")))

(file-system
    (device "/dev/mapper/home")
    (mount-point "/home")
    (type "ext3")))
Status.
- Started in 2012
- **4,400+ packages**, all free software
- **4 architectures**:
  - x86_64, i686, ARMv7, mips64el
- Binaries at [https://hydra.gnu.org](https://hydra.gnu.org)
- 0.11.0 released in August 2016
cluster deployments & usage

- Max Delbrück Center (DE): 250-node cluster + workstations
- Utrecht Bioinformatics Center (NL): 68-node cluster (1,000+ cores)
- GeneNetwork, "framework for web-based genetics"
30 Day Summary
Oct 4 2016 — Nov 3 2016
630 Commits
41 Contributors
  including 5 new contributors

12 Month Summary
Nov 3 2015 — Nov 3 2016
6490 Commits
  Up + 1434 (28%) from previous 12 months

106 Contributors
  Up + 53 (100%) from previous 12 months
Wrap-up.
Guix supports **reproducible software environments**

... can be extended with **personal packages**

... allows for **experimentation** through customization

... is entirely **programmable**
GuixSD

ludo@gnu.org

https://gnu.org/software/guix/