Reproducible builds ecosystem

Where some of us are
and some hints where this might be going...

Holger 'h01ger' Levsen
holger@layer-acht.org

FOSDEM16 (Brussels, Belgium)
2016-01-31

about me

- B8BF 5413 7B09 D35C F026 FE9D 091A B856 069A AA1C
- Debian user since 1995
- Debian contributor since 2001
- Debian developer since 2007
- DebConf organizer, founded the DebConf video team
  - http://video.debian.net
- Debian-Edu (Debian for education)
- Debian QA (quality assurance)
  - https://piuparts.debian.org
  - https://jenkins.debian.net (1100 jobs continously testing Debian)
- Debian LTS (Long Term Support)

about me

- B8BF 5413 7B09 D35C F026 FE9D 091A B856 069A AA1C
- Debian user since 1995
- Debian contributor since 2001
- Debian developer since 2007
- ~~DebConf organizer~~, founded the DebConf video team
  - http://video.debian.net
- ~~Debian-Edu~~ (Debian for education)
- Debian QA (quality assurance)
  - https://piuparts.debian.org
  - https://jenkins.debian.net (1100 jobs continously testing Debian)
- ~~Debian LTS~~ (Long Term Support)

about me

- B8BF 5413 7B09 D35C F026 FE9D 091A B856 069A AA1C
- Debian user since 1995
- Debian contributor since 2001
- Debian developer since 2007
- DebConf organizer, founded the DebConf video team
  - http://video.debian.net
- Debian-Edu (Debian for education)
- Debian QA (quality assurance)
  - https://piuparts.debian.org
  - https://jenkins.debian.net (1100 jobs continously testing Debian)
- Debian LTS (Long Term Support)

Detour: naming is a hard problem…

- `https://jenkins.debian.net`
- `https://reproducible.debian.net`
- `https://tests.reproducible-builds.org`
- `https://reproducible-builds.org`

Detour: naming is a hard problem…

- `https://jenkins.debian.net`
- ~~`https://reproducible.debian.net`~~
- `https://tests.reproducible-builds.org`
- `https://reproducible-builds.org`
- all point to 78.137.96.196

Detour: naming is a hard problem…

- `https://jenkins.debian.net`
- ~~`https://reproducible.debian.net`~~
- `https://tests.reproducible-builds.org`
- `https://reproducible-builds.org`
- all point to `78.137.96.196`
- and sometimes only `https://tests.r-b.org` fit on the slides

more about me

- B8BF 5413 7B09 D35C F026 FE9D 091A B856 069A AA1C
- 8F03 B243 8719 BA6B 1A35 0EB6 40C2 DEA2 F56C 7256
- Debian Reproducible builds team member
  - until April 2016 together with Lunar funded by the Linux Foundation
  - within in the team I'm mostly working on https://tests.reproducible-builds.org

more about me

- B8BF 5413 7B09 D35C F026 FE9D 091A B856 069A AA1C
- 8F03 B243 8719 BA6B 1A35 0EB6 40C2 DEA2 F56C 7256
- Debian Reproducible builds team member
  - until April 2016 together with Lunar funded by the Linux Foundation
  - within in the team I'm mostly working on https://tests.reproducible-builds.org
- sudo apt-get install torbrowser-launcher

Debian reproducible builds team

akira
Andrew Ayer
Asheesh Laroia
Chris Lamb
Chris West
Christoph Berg
Daniel Kahn Gillmor
David Suarez
Dhole
Drew Fisher
Esa Peuha
Guillem Jover

Hans-Christoph
Steiner
Helmut Grohne
Holger Levsen
Jelmer Vernooij
josch
Juan Picca
Lunar
Mathieu Bridon
Mattia Rizzolo
Nicolas Boulenguez
Niels Thykier
Niko Tyni

Paul Wise
Peter De Wachter
Philip Rinn
Reiner Herrmann
Stefano Rivera
Stéphane Glondu
Steven Chamberlain
Tom Fitzhenry
Valentin Lorentz
Wookey
Ximin Luo

Debian reproducible builds team

akira
Andrew Ayer
Asheesh Laroia
Chris Lamb
Chris West
Christoph Berg
Daniel Kahn Gillmor
David Suarez
Dhole
Drew Fisher
Esa Peuha
Guillem Jover

Hans-Christoph
Steiner
Helmut Grohne
Holger Levsen
Jelmer Vernooij
josch
Juan Picca
Lunar
Mathieu Bridon
Mattia Rizzolo
Nicolas Boulenguez
Niels Thykier
Niko Tyni

Paul Wise
Peter De Wachter
Philip Rinn
Reiner Herrmann
Stefano Rivera
Stéphane Glondu
Steven Chamberlain
Tom Fitzhenry
Valentin Lorentz
Wookey
Ximin Luo

jenkins.debian.net.git contributors

akira
Alexander Couzens
Levente 'anthraxx' Polyak
Antonio Terceiro
Axel Beckert
Bryan Newbold
Chris Lamb
Daniel Kahn Gillmor
Gabriele Giacone
Hans-Christoph Steiner
Helmut Grohne
Holger Levsen
HW42
James McCoy
Joachim Breitner

Johannes 'josch' Schauer
Jérémy Bobbio
Mattia Rizzolo
Niels Thykier
Paul Wise
Petter Reinholdtsen
Philip Hands
Reiner Herrmann
Samuel Thibault
Steven Chamberlain
Tails developers
Ulrike Uhlig
Wolfgang Schweer
Wouter Verhelst

jenkins.debian.net.git contributors

akira
Alexander Couzens
Levente 'anthraxx' Polyak
Antonio Terceiro
Axel Beckert
Bryan Newbold
Chris Lamb
Daniel Kahn Gillmor
Gabriele Giacone
Hans-Christoph Steiner
Helmut Grohne
Holger Levsen
HW42
James McCoy
Joachim Breitner

Johannes 'josch' Schauer
Jérémy Bobbio
Mattia Rizzolo
Niels Thykier
Paul Wise
Petter Reinholdtsen
Philip Hands
Reiner Herrmann
Samuel Thibault
Steven Chamberlain
Tails developers
Ulrike Uhlig
Wolfgang Schweer
Wouter Verhelst

Who are you?

- Contributed to Free Software?

Who are you?

- Contributed to Free Software?
- Seen a talk about reproducible builds last year?

Who are you?

- Contributed to Free Software?
- Seen a talk about reproducible builds last year?
- Contributed to this effort?

The problem
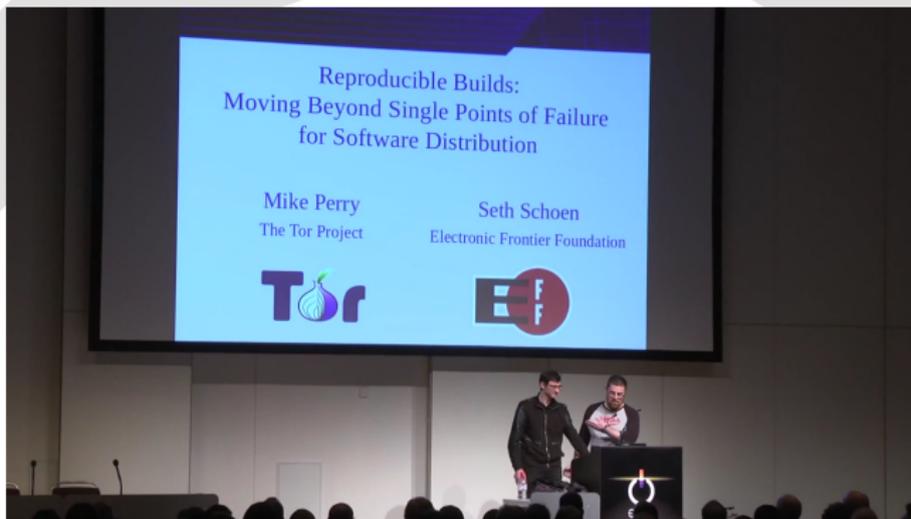
Available on `media.ccc.de`, 31c3

A few examples from that 31c3 talk

- CVE-2002-0083: remote root exploit in `sshd`, a single bit difference in the binary

A few examples from that 31c3 talk

- CVE-2002-0083: remote root exploit in `sshd`, a single bit difference in the binary
- 31c3 talk had a live demo with a kernel module modifying source code in `memory` only

A few examples from that 31c3 talk

- CVE-2002-0083: remote root exploit in `sshd`, a single bit difference in the binary
- 31c3 talk had a live demo with a kernel module modifying source code in memory only
- financial incentives to crack developer machines…

A few examples from that 31c3 talk

- CVE-2002-0083: remote root exploit in `sshd`, a single bit difference in the binary
- 31c3 talk had a live demo with a kernel module modifying source code in memory only
- financial incentives to crack developer machines…
- how can you be sure what's running on your machine or on a build daemon network? Do you ever leave your USB3 ports alone?

A few examples from that 31c3 talk

- CVE-2002-0083: remote root exploit in `sshd`, a single bit difference in the binary
- 31c3 talk had a live demo with a kernel module modifying source code in `memory` only
- financial incentives to crack developer machines...
- how can you be sure what's running on your machine or on a build daemon network? Do you ever leave your computers alone?

Another example from real life

At a CIA conference in 2012:

```
firstlook.org/theintercept/2015/03/10/

ispy-cia-campaign-steal-apples-secrets/
```

The solution

Promise that anyone can always generate identical binary packages from a given source

The solution

We call this:

"Reproducible builds"

Demo
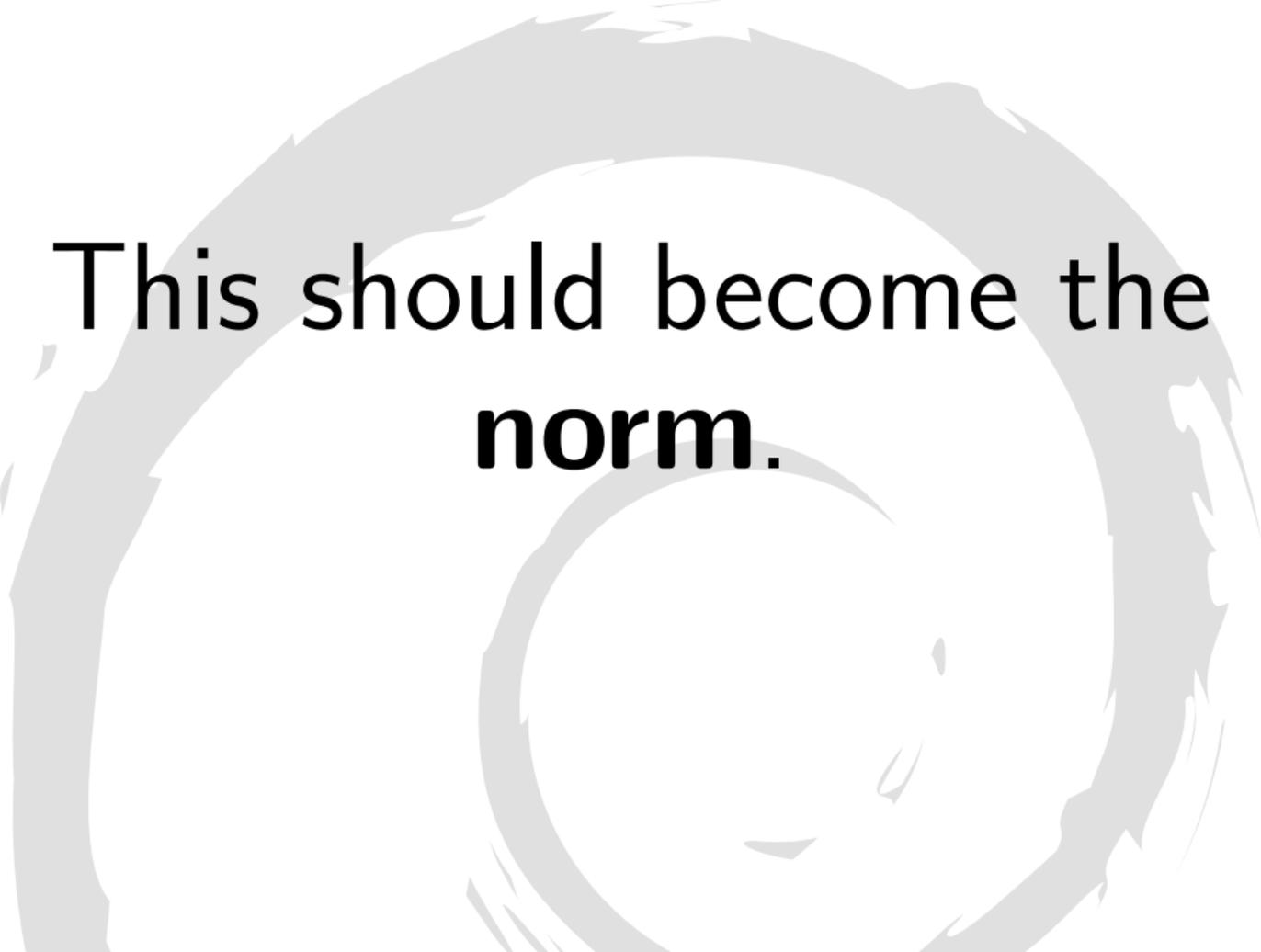
This should become the **norm**.
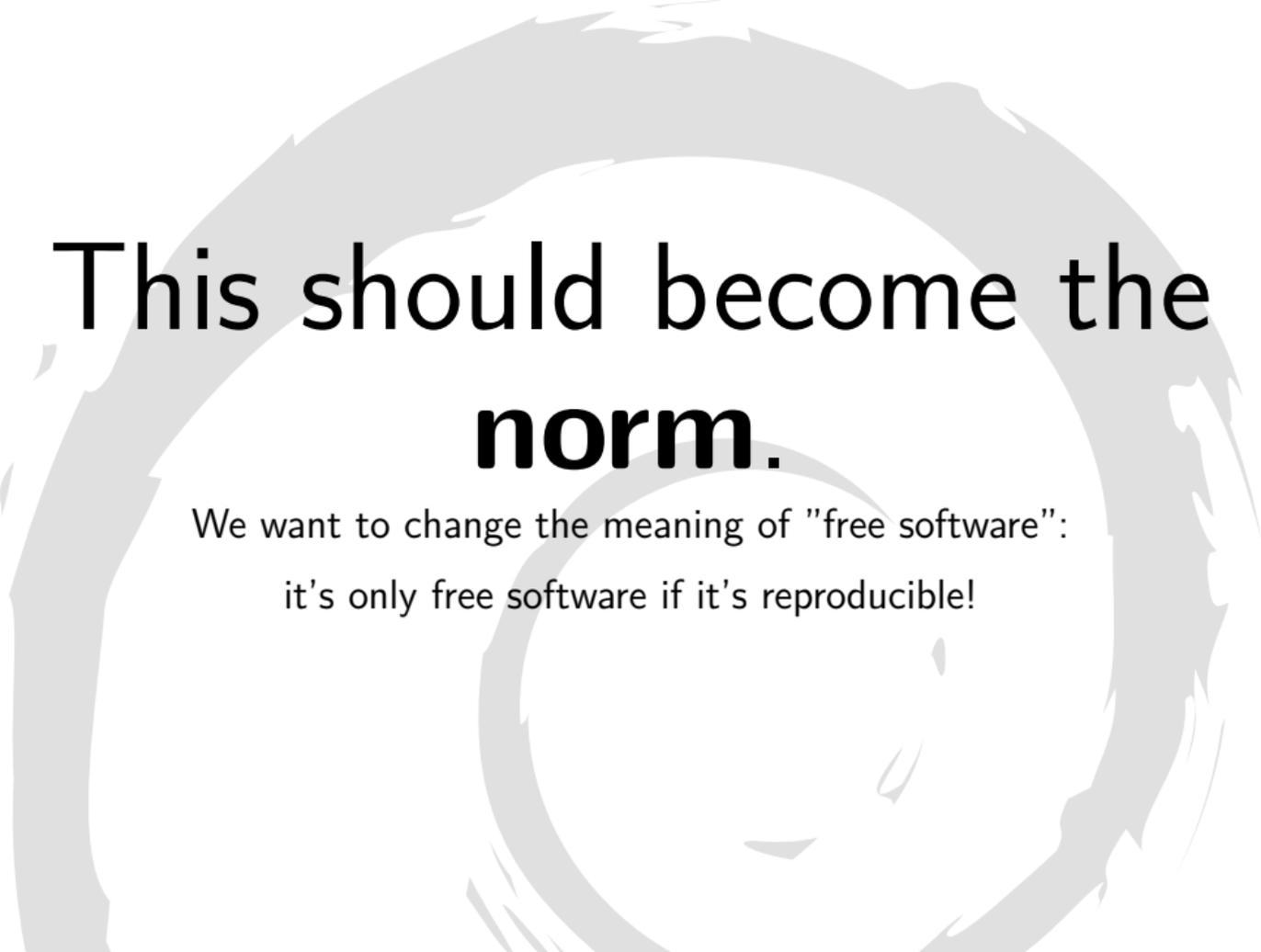
This should become the **norm**.

We want to change the meaning of "free software":

it's only free software if it's reproducible!

reproducible-builds.org

- `https://reproducible-builds.org`

reproducible-builds.org

Provide a verifiable path from source code to binary.

What is it about?

Reproducible builds are a set of software development practices which create a **verifiable path from** human readable **source code to** the **binary** code used by computers.

Why does it matter?

Most aspect of software verification is done on source code, as that is what humans can reasonably understand. But most of the time, computers require software to be first built into long string of numbers to be used. With *reproducible builds*, multiple parties can **redo this process independently** and ensure they **all get** *exactly* **the same result**. We can thus **grow confidence** than a

Documentation about common problems

- `https://reproducible-builds.org/docs`
- Lunar's talk from CCCamp 2015 also on `https://media.ccc.de`
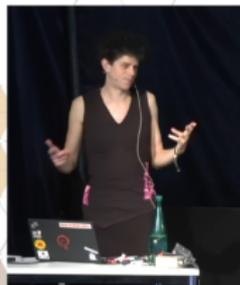
Common problems

- time stamps

Common problems

- time stamps
- timezones
- locales

Common problems

- time stamps
- timezones
- locales
- everything else (seperated into known issues and the blurry rest)

SOURCE_DATE_EPOCH

- Build date (timestamps) usually not useful for the user

SOURCE_DATE_EPOCH

- Build date (timestamps) usually not useful for the user
- SOURCE_DATE_EPOCH is defined as the last modification of the source, since the epoch (1970-01-01)
- SOURCE_DATE_EPOCH can be used instead of current date
- can also be used for random seeds etc.

SOURCE_DATE_EPOCH

- Build date (timestamps) usually not useful for the user
- SOURCE_DATE_EPOCH is defined as the last modification of the source, since the epoch (1970-01-01)
- SOURCE_DATE_EPOCH can be used instead of current date
- can also be used for random seeds etc.
- in Debian, set from the latest debian/changelog entry
- solution has been adopted by other projects & distributions (NetBSD, FreeBSD, Arch Linux, Guix, …)

SOURCE_DATE_EPOCH

- SOURCE_DATE_EPOCH spec availble
- https://reproducible-builds.org/specs/

SOURCE_DATE_EPOCH (closed bugs)

- dh-strip-nondeterminism
- #~~791823~~: debhelper
- #~~787444~~: help2man
- #~~790899~~: epydoc
- #~~794004~~: ghostscript
- #~~796130~~: man2html
- #~~783475~~: texi2html
- #~~794586~~: ocamldoc
- #~~795942~~: wheel
- …

SOURCE_DATE_EPOCH (open/pending bugs)

- gcc (__DATE__ and __TIME__ macros)
  https://gcc.gnu.org/ml/gcc-patches/2015-06/msg02210.html
- #792687, #804141: gettext
- #792201: doxygen
- #800797: docbook-utils
- #801621: perl (Pod::Man)
- #790801: txt2man
- #791815: libxslt
- #794681: qt4-x11 (qthelpgenerator)
- #792202: texlive-bin
- ...

tests.reproducible-builds.org

- Continuously testing Debian `testing`, `unstable` and `experimental`
- Also testing: coreboot, OpenWrt, NetBSD, FreeBSD, Arch Linux, Fedora and soon F-Droid and Guix too

tests.reproducible-builds.org

- Continuously testing Debian `testing`, `unstable` and `experimental`
- Also testing: coreboot, OpenWrt, NetBSD, FreeBSD, Arch Linux, Fedora and soon F-Droid and Guix too
- 230 jenkins jobs running on 22 hosts
- 42 scripts with a total of 4k lines of Python and 6k lines of Bash Shell
- 29 contributors for `jenkins.debian.net.git`

tests.reproducible-builds.org

- Continuously testing Debian `testing`, `unstable` and `experimental`
- Also testing: coreboot, OpenWrt, NetBSD, FreeBSD, Arch Linux, Fedora and soon F-Droid and Guix too
- 230 jenkins jobs running on 22 hosts
- 42 scripts with a total of 4k lines of Python and 6k lines of Bash Shell
- 29 contributors for `jenkins.debian.net.git`
- Really simple code.
- Untested patches are fine as long as they only break your stuff.

Variations (when testing Debian)

variation	first build	second build
hostname	`jenkins`	`i-capture-the-hostname`
domainname	`debian.net`	`i-capture-the-domainname`
env `TZ`	`GMT+12`	`GMT-14`
env `LANG`	`C`	`fr_CH.UTF-8`
env `LC_ALL`	not set	`fr_CH.UTF-8`
env `USER`	`pbuilder1`	`pbuilder2`
uid	1111	2222
gid	1111	2222
UTS namespace	shared with the host	*modified using /usr/bin/unshare --uts*
kernel version	Linux 3.16 or 4.X	on amd64 always varied, on armhf sometimes
umask	0022	0002
CPU type	same for both builds on amd64 *(work in progress)* on armhf varied a bit	
filesystem	same for both builds on amd64: (`tmpfs`), on armhf ext3/4 *(and we have* `disorderfs`*, but the code is disabled)*	
year, month, date	on amd64: 398 days variation, on armhf not yet	
hour, minute	hour is usually the same… usually, the minute differs…	
everything else	*is likely the same…*	

Debugging problems: `diffoscope`

- Examines differences **in depth**.
- Outputs HTML or plain text with human readable differences.
- Recursively unpacks archives, uncompresses PDFs, disassembles binaries, unpacks Gettext files, …
- Easy to extend to new file formats.
- Falls back to binary comparison.
- Available from `git`, PyPI, Debian (sid and stretch), Arch Linux, Guix, Homebrew. Works on BSD.
- Maintainers in other distros wanted.
- `https://diffoscope.org/`
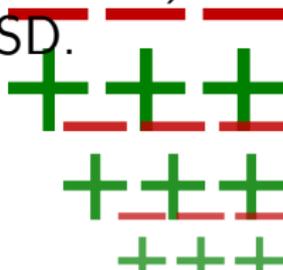
diffoscope example (HTML output)

Try diffoscope

- https://try.diffoscope.org

diffoscope is "just" for debugging

- Reminder: `diffoscope` is for **debugging**

diffoscope is "just" for debugging

- Reminder: `diffoscope` is for **debugging**
- "reproducible" according to our definition means: **bit by bit identical**. So the tools for testing whether something is reproducible are either `diff` or `sha256sum`!
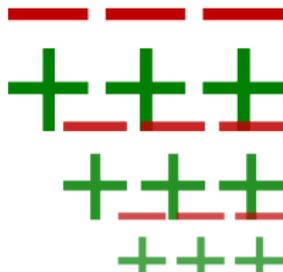
amd64 architecture on tests.r-b.org

- amd64: 106 cores and 282 GB RAM split on 9 VMs
  - 4 VMs with 17/18 cores and 32 GB for Debian
  - 1 VM (soon 2) with 8 cores and 16 GB for most of the rest
  - 1 jenkins VM and 1 jenkins-test VM
  - 1 extra VM for building FreeBSD

- sponsored by `https://profitbricks.co.uk`

armhf architecture on tests.r-b.org

- `armhf`: 14 nodes with 50 cores and 23 GB RAM
  - ▸ a few more nodes planned
  - ▸ combined they only draw 100 watts of power

- hosted by `vagrant@debian.org`, several hardware sponsors (Debian, TechNexion, LeMaker, SolidRun, Beagleboard.org and Aikidev) plus 4.5k USD by Debian
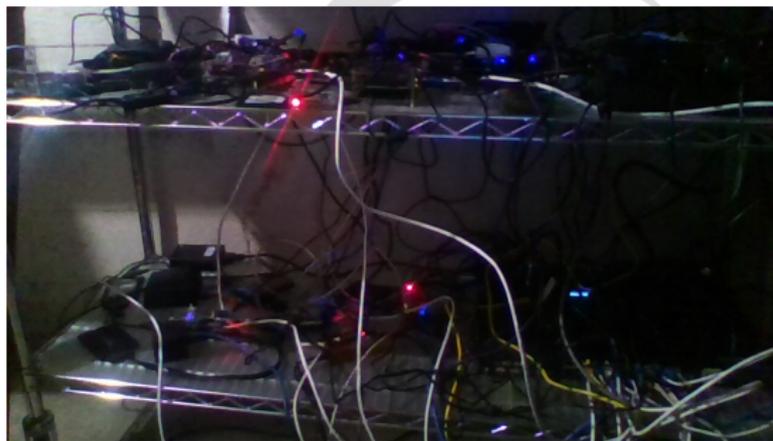
armhf architecture on tests.r-b.org

- `armhf`: 14 nodes with 50 cores and 23 GB RAM
  - ▸ a few more nodes planned
  - ▸ combined they only draw 100 watts of power
  - ▸ roughly takes 4 weeks to build one Debian suite
- hosted by `vagrant@debian.org`, several hardware sponsors (Debian, TechNexion, LeMaker, SolidRun, Beagleboard.org and Aikidev) plus 4.5k USD by Debian
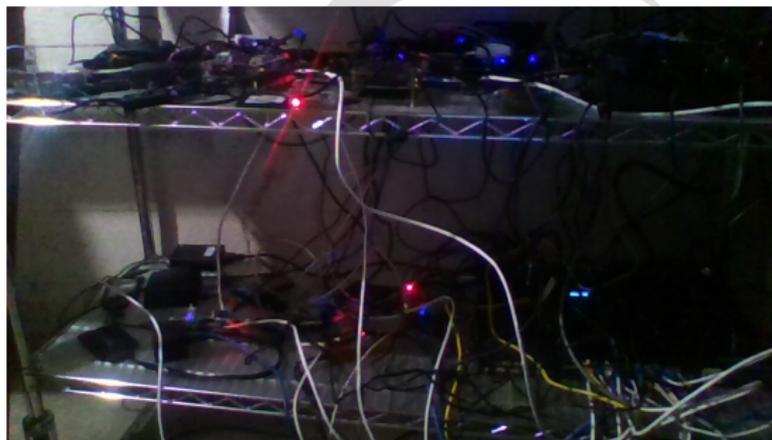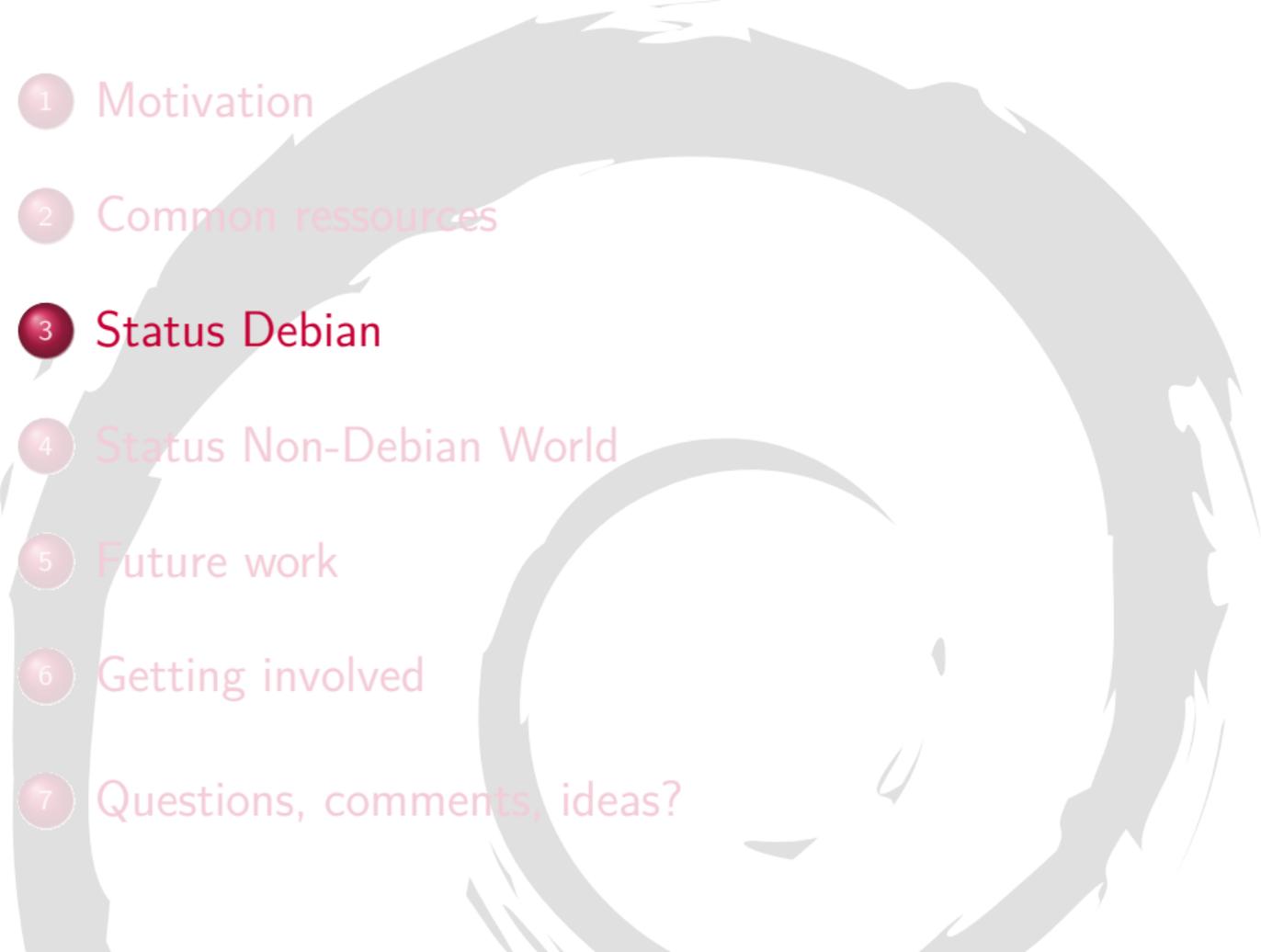
Progress in Debian `unstable`

Reproducibility status for packages in 'unstable' for 'amd64'
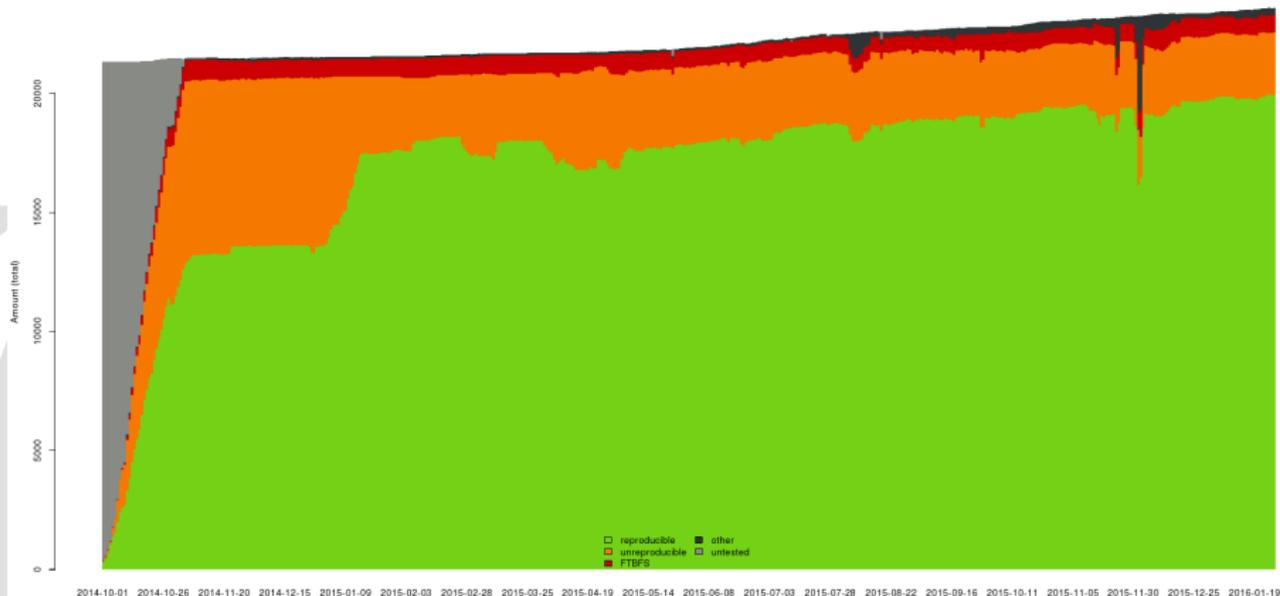
20,079 (85%) out of 23,595 source packages are reproducible
in our test framework

Notes and issues on tests.reproducible-builds.org

- 179 categorised distinct issues
- 3,792 notes

Notes and issues on tests.reproducible-builds.org

- 179 categorised distinct issues
- 3,792 notes
- 2549 unreproducible packages in `sid`, but only 139 without a note
- 728 packages failing to build, but only 74 without a note

Notes and issues on tests.reproducible-builds.org

- 179 categorised distinct issues
- 3,792 notes
- 2549 unreproducible packages in `sid`, but only 139 without a note
- 728 packages failing to build, but only 74 without a note
- maintained in `notes.git`

Notes and issues on tests.reproducible-builds.org

- 179 categorised distinct issues
- 3,792 notes
- 2549 unreproducible packages in `sid`, but only 139 without a note
- 728 packages failing to build, but only 74 without a note
- maintained in `notes.git`
- currently Debian only, but cross distro notes are planned

Debian packages on tests.reproducible-builds.org

- `https://reproducible.debian.net/$src`

Debian package sets on tests.r-b.org

Reproducibility status for packages in unstable from 'required'
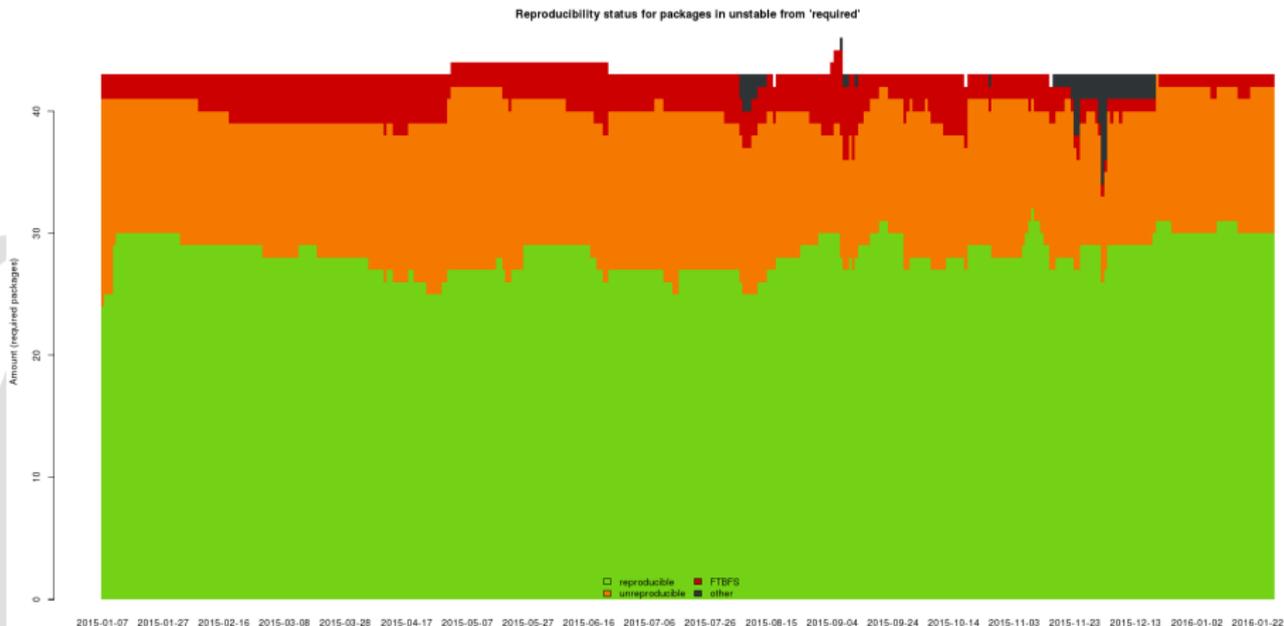
29 different "package sets", eg. `required` is only 68% reproducible

Progress in the Debian bug tracker

Open and closed bugs (with all usertags except tagged 'ftbfs')
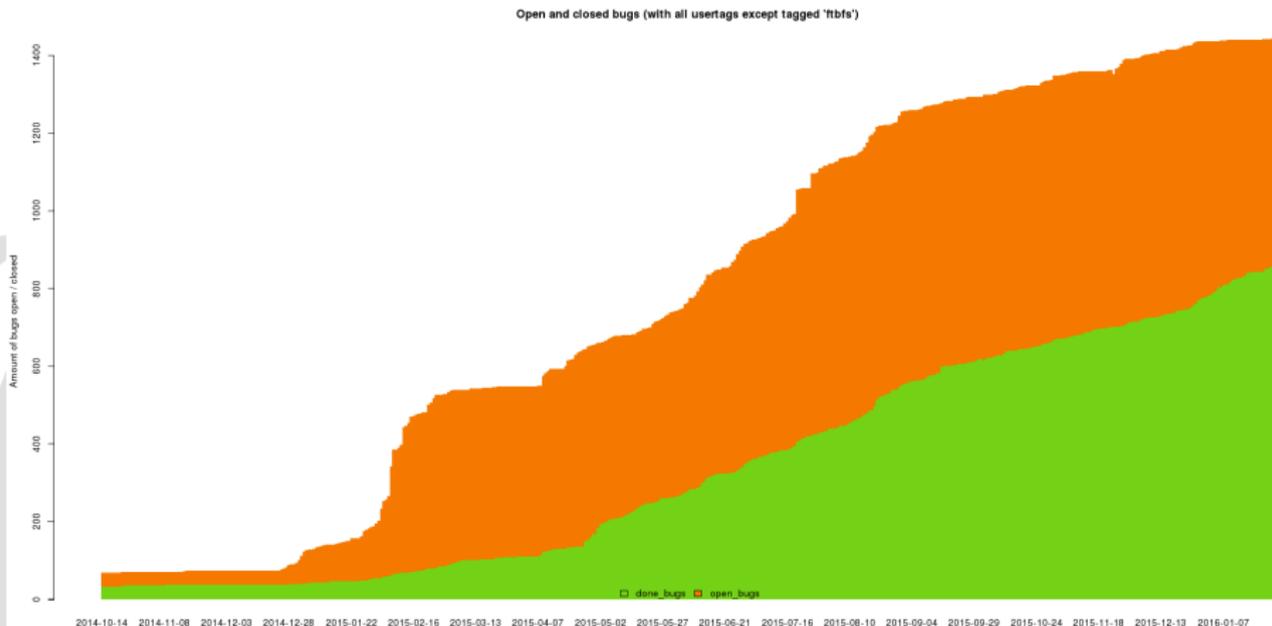
As a rule, we file bugs with patches.
There are very few exceptions.

What we did in Debian

- Agreed on using a fixed build path: `/build/`
- Recording the build environment: `.buildinfo`
- `strip-nondeterminism`
- `diffoscope` (formerly `debbindiff`)
- `SOURCE_DATE_EPOCH`
- `disorderfs`
- 1000+ patches: `dpkg`, `debhelper`, `sbuild`, …
- …

Tell the world & collaborate

- Weekly reports since May 2015

Tell the world & collaborate

- Weekly reports since May 2015
- First Reproducible World Summit in December 2015 (Athens, Greece)
  - 40 people from 16 projects
  - `reproducible.debian.net` has become `tests.reproducible-builds.org`

Tell the world & collaborate

- Weekly reports since May 2015
- First Reproducible World Summit in December 2015 (Athens, Greece)
  - ▸ 40 people from 16 projects
  - ▸ `reproducible.debian.net` has become `tests.reproducible-builds.org`
  - ▸ another Summit in April/May 2016?!

Tell the world & collaborate

- Weekly reports since May 2015
- First Reproducible World Summit in December 2015 (Athens, Greece)
  - ▶ 40 people from 16 projects
  - ▶ `reproducible.debian.net` has become `tests.reproducible-builds.org`
  - ▶ another Summit in April/May 2016?!
- 2 GSoC students in 2015, totally new contributors, totally rocking
  - ▶ more GSoC/Outreachy contributors this year?!!

Detour: Reproducible builds demand a defined build environment

- …and being able to re-create this build environment is mandatory too.
- Without an *sufficiently identical* build environment, reproducible builds will only happen by sheer luck.

Detour: Reproducible builds demand a defined build environment

- …and being able to re-create this build environment is mandatory too.
- Without an *sufficiently identical* build environment, reproducible builds will only happen by sheer luck.
- I've only seen this verified for Debian so far… `koji` is designed for that too, Guix as well…
- I'd very much like to be corrected here, with tests.

Debian `.buildinfo` files

- Aggregates in the same file:
  - Sources (checksums)
  - Generated binaries (checksums)
  - Packages used to build (with specific version, checksums coming soon)
- Can be later used to exactly recreate environment
- For Debian, all versions are available from `snapshot.debian.org`

Example `.buildinfo` file

```
Format: 1.9
Build-Architecture: amd64
Source: txtorcon
Binary: python-txtorcon
Architecture: all
Version: 0.11.0-1
Build-Path: /build/txtorcon-0.11.0-1
Checksums-Sha256:
 a26549d9…7b 125910 python-txtorcon_0.11.0-1_all.deb
 28f6bcbe…69 2039 txtorcon_0.11.0-1.dsc
Build-Environment:
 base-files (= 8),
 base-passwd (= 3.5.37),
 bash (= 4.3-11+b1),
 …
```

Blockers for Debian: `dpkg` and `dak`

- `dpkg`
  - #138409: dpkg-dev: please add support for .buildinfo files
  - ~~#719844: make compression of {data,control}.tar.gz deterministic~~
  - #759999: set reproducible timestamps in `.deb` ar file headers
  - #787980: normalize file permissions when creating control.tar
  - #719845: make file order within data,control.tar.gz deterministic
- `dak` (`ftp.debian.org`)
  - #763822: please include `.buildinfo` files in the archive

`debian-policy`

- Section 4.15: "Sources **must** build reproducible binaries."

`debian-policy`

- Section 4.15: "Sources **must** build reproducible binaries."
- We hope this will happen after stretch (Debian 9) release

`debian-policy`

- Section 4.15: "Sources **must** build reproducible binaries."
- We hope this will happen after stretch (Debian 9) release
- In 2016: "Sources **shall** build reproducible binaries."

Reminder / Summary

- This is just a proof-of-concept, Debian is not 85% reproducible
- Patches still need to be merged

Reminder / Summary

- This is just a proof-of-concept, Debian is not 85% reproducible
- Patches still need to be merged
- I hope that Debian 9, "stretch", will be partially reproducible in a meaningful way

Reminder / Summary

- This is just a proof-of-concept, Debian is not 85% reproducible
- Patches still need to be merged
- I hope that Debian 9, "stretch", will be partially reproducible in a meaningful way
- Debian "unstable" as an easter (=end of March 2016) present?

Reminder / Summary

- This is just a proof-of-concept, Debian is not 85% reproducible
- Patches still need to be merged
- I hope that Debian 9, "stretch", will be partially reproducible in a meaningful way
- Debian "unstable" as an easter (=end of March 2016) present?
- what's beyond (rebuilding, `.buildinfo` file handling, user tools) mostly still needs *design and code*
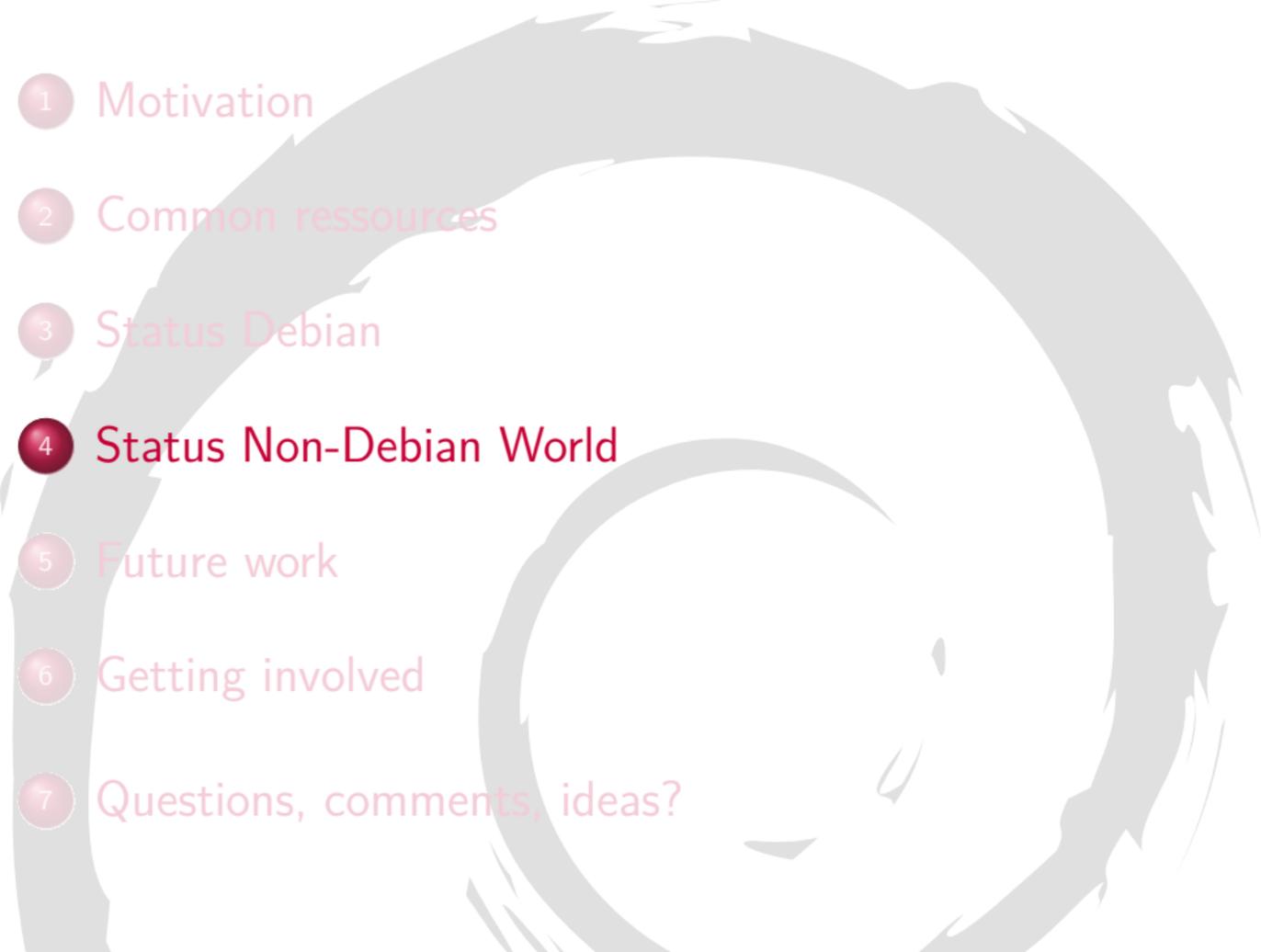
Status coreboot

- https://tests.r-b.org/coreboot
- 99.2% reproducible with `seabios` payload
- tests maintained by Alexander 'lynxis' Couzens
- unclear what the next steps are...
- needs more active involvement from coreboot developers

Status OpenWrt

- https://tests.r-b.org/coreboot
- selected images are 100% reproducible and selected packages 99.7%
- using 13 patches send upstream on January 25th
- tests maintained by Alexander 'lynxis' Couzens and Bryan Newbold
- recreating the build env: needs to checked in practice
- user verification tools: not yet
- next, once patches are merged: rebuilding released binaries?!

Status NetBSD

- https://tests.r-b.org/netbsd
- 21 (38.8%) out of 54 built NetBSD files are reproducible
- tests maintained by Thomas 'wiz' Klausner and h01ger
- MKREPRO=yes
- MK_TIMESTAMP=$SOURCE_DATE_EPOCH
- recreating the build env: ?
- user verification tools: not yet
- next: ask Thomas :)

Status FreeBSD

- https://tests.r-b.org/freebsd
- base system not yet reproducible, but almost there
- 63% of 15k ports were reproducible in 2013 already, their wiki says
- tests maintained by h01ger
- recreating the build env: ?
- user verification tools: not yet
- talk today, in K.4.601 at 15:40 CET
- next: watch talks, discuss with develope

Status ElectroBSD

- I have no idea... but am curious to find out!
- talk today, in K.4.601 at 14:35 CET

Status Fedora

- https://tests.r-b.org/fedora (23)
- maintained by Dhiru Kholia and h01ger
- rpm repo available by Dhiru, but still **0% reproducible**
- rpm format includes build time and build host and…
- recreating the build env: koji
- next: 24+rawhide, first reproducible rpm
- next: get more people involved

Status Arch Linux

- `https://tests.r-b.org/archlinux`
- maintained by Levente 'anthraxx' Polyak and h01ger
- reproducible patches available for `pacman` by anthraxx
- recreating the build env: unaddressed
- next: use those patches, upstream them

Status F-Droid

- not yet: `https://tests.r-b.org/f-droid`
- maintained by Hans-Christoph Steiner and h01ger
- work has just begun…

Status Guix

- I still have little idea, sadly missed the talk yesterday
- recreating the build env: by design
- user verification tool: yes! (Guix challange)

Status Guix

- I still have little idea, sadly missed the talk yesterday
- recreating the build env: by design
- user verification tool: yes! (Guix challange)
- next: check your inbox, h01ger!

Unmentioned, with known activities

- Bitcoin
- Tor
- NixOS
- Qubes
- commercial, propietary Software
- ?

Detour: what, reproducible commercial Software???

- Guess which

Detour: what, reproducible commercial Software???

- Guess which
- windows? (the source is available)
- medical devices in your body?
- arms?
- critical infrastructure like in nuclear powerplants?
- cars?

Detour: what, reproducible commercial Software???

- Guess which
- windows? (the source is available)
- medical devices in your body?
- arms?
- critical infrastructure like in nuclear powerplants?
- cars?
- Gambling machines!

Unmentioned, unknown activities?

- OpenSUSE (could be tested easily…)
- Ubuntu
- OpenBSD
- Gentoo (stage1)
- ?
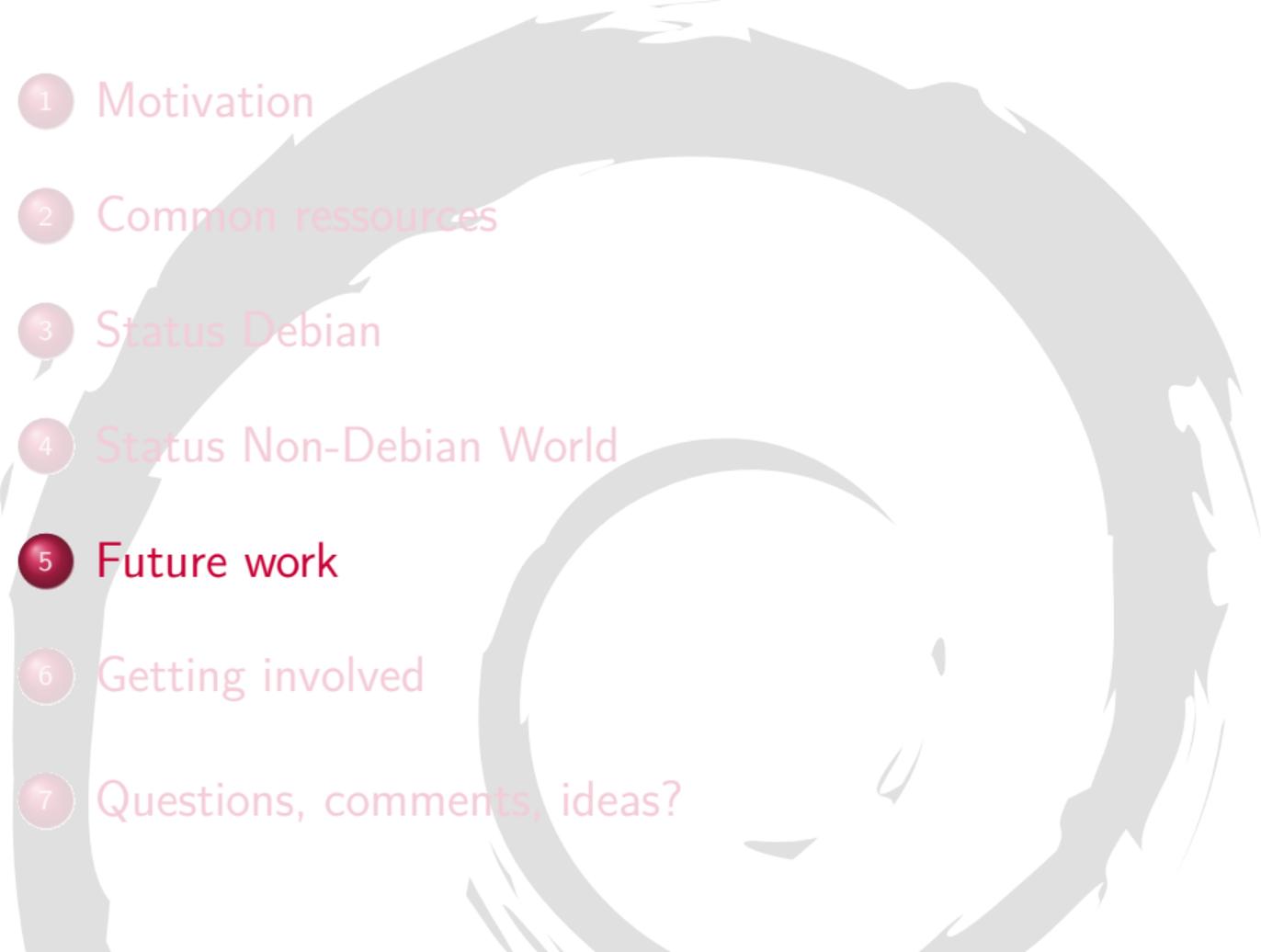
Release process issues

- In our current design and practices, rebuilding Debian `stretch` will require package versions which are not part of `stretch`.
- Rebuilding all of Debian a month prio the release?

Release process issues

- In our current design and practices, rebuilding Debian `stretch` will require package versions which are not part of `stretch`.
- Rebuilding all of Debian a month prio the release?
- This problem is per se not Debian specific at all...

Distributing `.buildinfo` files

- Probably 100,000 new files per Debian suite; 50% increase per suite
- Mirrors would not be happy, so should not go there

Distributing `.buildinfo` files

- Probably 100,000 new files per Debian suite; 50% increase per suite
- Mirrors would not be happy, so should not go there
- We'll need more files with detached signatures...
- Revoking signatures?
- ...

Rebuilders and sharing signed checksums

- Almost no work has been done here yet.

Rebuilders and sharing signed checksums

- Almost no work has been done here yet.
- Systematic, automatic rebuilds?

Rebuilders and sharing signed checksums

- Almost no work has been done here yet.
- Systematic, automatic rebuilds?
- Different projects, different solutions?

Rebuilders and sharing signed checksums, cont.

- Individuelly signed checksums (think web of trust) could work in the Debian case (we have a gpg web of trust), but IMO won't scale.

Rebuilders and sharing signed checksums, cont.

- Individuelly signed checksums (think web of trust) could work in the Debian case (we have a gpg web of trust), but IMO won't scale.
- Another idea: rebuilders, run by large organisations (ACLU, CCC, CERN, Deutsche Bank, EDF, EON, Greenpeace, NASA, NSA, XYZ).

Rebuilders and sharing signed checksums, cont.

- Individuelly signed checksums (think web of trust) could work in the Debian case (we have a gpg web of trust), but IMO won't scale.
- Another idea: rebuilders, run by large organisations (ACLU, CCC, CERN, Deutsche Bank, EDF, EON, Greenpeace, NASA, NSA, XYZ).
- Fedora rebuilds Debian, Debian rebuilds OpenSUSE, OpenSUSE rebuilds NetBSD, etc…

Integration in user tools

- "Do you really want to install this unreproducible software (y/N)"

Integration in user tools

- "Do you really want to install this unreproducible software (y/N)"
- "Do you want to build those packages which unconfirmed checksums, before installing? (Y/n)"

Integration in user tools

- "Do you really want to install this unreproducible software (y/N)"
- "Do you want to build those packages which unconfirmed checksums, before installing? (Y/n)"
- "How many signed checksums do you require to call a package 'reproducible'?"

Integration in user tools

- "Do you really want to install this unreproducible software (y/N)"
- "Do you want to build those packages which unconfirmed checksums, before installing? (Y/n)"
- "How many signed checksums do you require to call a package 'reproducible'?"
- "Which rebuilders do you want to trust?"

Future of tests.reproducible-builds.org

- We still want more (arm(64)) cores!
- We want to test other architectures!
- We want to test other projects!
- We want more people contributing code for their projects!
- We want more people looking at the results!
- We don't want to build twice and test against what we built, but rather the binaries distributed by these projects (if any)

Future of tests.reproducible-builds.org

- We still want more (arm(64)) cores!
- We want to test other architectures!
- We want to test other projects!
- We want more people contributing code for their projects!
- We want more people looking at the results!
- We don't want to build twice and test against what we built, but rather the binaries distributed by these projects (if any)
- This is "just" a testing framework...

Summary

- We've come a long way.
- We've made impressive progress.
- We're still not nearly where we want to be.

Summary

- We've come a long way.
- We've made impressive progress.
- We're still not nearly where we want to be.
- In fact, it's still fully not clear where we need to be going.

Summary

- We've come a long way.
- We've made impressive progress.
- We're still not nearly where we want to be.
- In fact, it's still fully not clear where we need to be going.
- Keep up the great work!
- Join the fun! There are many big and small things to do!
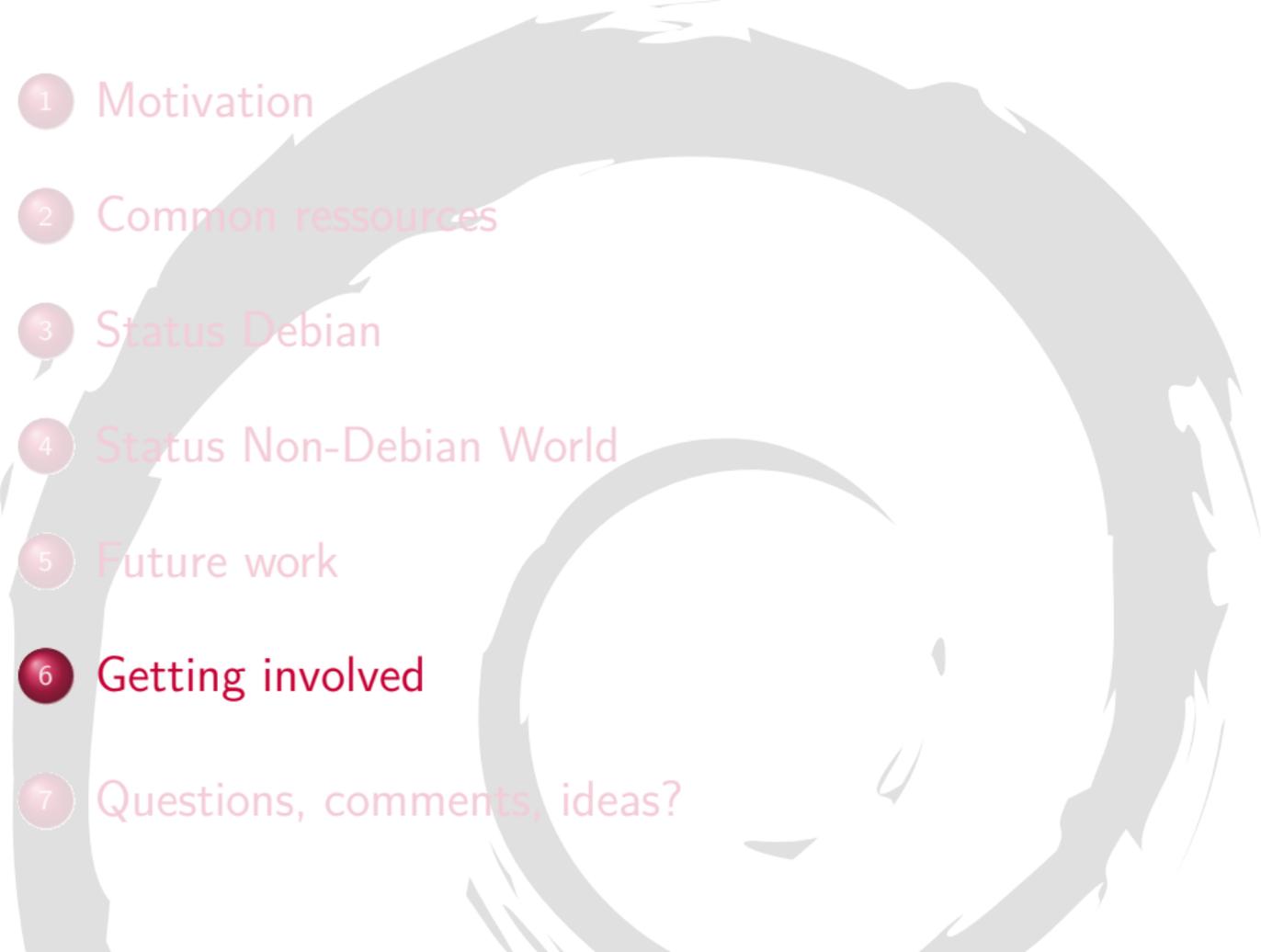
As a software developer

- Stop using build dates
- Use `SOURCE_DATE_EPOCH` instead
- See `https://reproducible-builds.org/specs/`

Getting involved - learning by doing

- Test for yourself:
  - ‣ Build something twice, run diffoscope on the results
    - ★ For better results use our "reproducible" repository, `pbuilder` and a custom config

- Docs on the web:
  `https://reproducible-builds.org/docs/`
  `https://wiki.debian.org/ReproducibleBuilds/`
  `ExperimentalToolchain`

- Ask for help on `#debian-reproducible` or on mailing list

Join the Reproducible builds team(s)!

- Why?
  - ♡♡♡ Lovely group of people ♡♡♡
  - Learn something new everyday
  - Change the (software) world!
- What do we do?
  - Review packages
  - Identify issues and document solutions
  - `reproducible.d.n`, diffoscope, strip-nondeterminism
  - Propose changes for toolchain
  - Submit patches for individual packages
  - Write more general documentation and talk to the world

Form another team!

- Why?
  - ▸ Every distribution should be reproducible!
  - ▸ Learn something new everyday
  - ▸ Change the (software) world!
  - ▸ https://tests.reproducible-builds.org/$project needs **your** help
- How to get started?
  - ▸ Talk to me here or talk to us on IRC or via mail.
  - ▸ RTFM, there is lots of documentation
  - ▸ Experiment - learning by doing

Listen to more talks in K.4.601 today!

- 14:35 ElectroBSD - Getting a reproducible BSD out of the door
- 15:40 Reproducible builds in FreeBSD packages

Questions, comments, ideas?

Questions, comments, ideas?

- https://reproducible-builds.org/docs
- https://tests.reproducible-builds.org
- #reproducible-builds on irc.OFTC.net
- and/or #debian-reproducible too!
- *with IRC notification by KGB-0*

Questions, comments, ideas?

- https://reproducible-builds.org/docs
- https://tests.reproducible-builds.org
- #reproducible-builds on irc.OFTC.net
- and/or #debian-reproducible too!
- *with IRC notification by KGB-0*
- https://lists.reproducible-builds.org
- https://twitter.com/ReproBuild

Thanks to…! …and thank **you**, too!

- Debian "Reproducible Builds" team
  (you are just **so** awesome!)
- Linux Foundation and the Core Infrastructure
  Initiative


```
holger@debian.org  B8BF 5413 7B09 D35C F026
                   FE9D 091A B856 069A AA1C
```

Thanks to...! ...and thank **you**, too!

- All "Reproducible Builds" teams
  (you are just **so** awesome!)
- Linux Foundation and the Core Infrastructure
  Initiative


```
holger@debian.org   B8BF 5413 7B09 D35C F026
                    FE9D 091A B856 069A AA1C
```