

News about Debian Autobuilding

Debconf 10

Philipp Kern

August 2, 2010

Table of contents

- 1 What did happen
- 2 What's next?
- 3 Current state
- 4 Questions / Comments

What did happen since last year?

- machine move
- edos-builddebcheck
- new architectures
- unification of software
- introduction of PostgreSQL instead of BDB
- new build priority system
- more suites

Machine move

- wanna-build host moved twice
- from raff at HP/Fort Collins to cimarosa, hosted by UBC
- machine unsuitable I/O-wise because of a missing RAID controller battery
- thus another move to grieg, hosted by GRNET
- more I/O bandwidth and more CPU power allowed us to take more tasks and finish our triggers quicker due to parallelization
- new machine is unrestricted, but do not use it for uber-data mining without consulting us first

Uninstallability checks

- Joachim Breitner's project of last year's Debconf
- integration uninstallability checks based on edos-builddebcheck into wanna-build

Uninstallability checks (2)

- introduction of a new state “BD-Uninstallable” before a package can be taken by a buildd (state “Needs-Build”)
- packages are generally only submitted to building after edos-debcheck assumes the build-dependencies to be installable
- on give-backs installability is rechecked
- serves as a much better automatic dep-wait than the old naive implementation
- some short rationale as given by edos-debcheck is left in the DB for inspection by developers
- dep-waits still exist for the cases that are not directly related to package installability (e.g. newer library needed to build the binNMU)

New architectures

- hurd-i386 moved to the official wanna-build instance
- its builddds still run separately

Unification of software

- all builddds now use the builddd branch of `git://git.debian.org/builddd-tools/sbuild.git`
- apt repository on `https://builddd.debian.org/apt/`
- builddd setup instructions on `https://builddd.debian.org/docs/builddd-setup.txt`
- new `builddd.debian.org` frontpage has links to source repositories

Introduction of PostgreSQL as the database backend

- *before*: Perl DBM hashes in BDB databases, only readable by Perl
- *now*: use of PostgreSQL for storing the package states and database changes
- porting done by Kurt Roeckx
- querying possible from any language, not just Perl
- no more file-based locks for read-only access, which hurt performance badly
- access restriction to the security databases is handled by using privileged psql users and views for the unprivileged users

Querying wanna-build

- on `buildd.debian.org` aka `grieg` you can query the live database via CLI
- *SQL interface*: use `psql service=wanna-build` to connect
- *wanna-build interface*:
 - `wanna-build` will use unprivileged PostgreSQL access for read-only queries
 - e.g. `wanna-build -A armel --list=needs-build`

New build priority system

- Andreas Barth adjusted the build priorities to not only consider the priority, section and component of a package
- ageing is applied to prevent starvation, a package's priority is increased over time to increase the likelihood of it being build
- used to ensure fair autobuilding on the backlogged mips,el autobuilders

More suites

- unofficial infrastructure (ayous.org) suffered from a machine compromise on the log host
- efforts were made not to resurrect it but to move the service to the official wanna-build instead
- quite a bit of pressure by fellow DDs to get the service back in time

More suites (2)

- backports, volatile, edu and experimental builds are now managed by the official wanna-build
- autobuilt by official builders where possible (requires LVM snapshots) and where capacities are available
- chroots are created for the base suites and rebranded by schroot setup scripts
- a new aptitude-based resolver was written by Marc Brockschmidt to ensure minimal build-dependencies in experimental

What about non-free?

- proper non-free autobuilding will probably be ready in one or two months
- building will be allowed against main, contrib and whitelisted non-free packages
- integration was harder because of the differentiation on components, which was not needed before available

Miscellaneous

- notifications of build failures sent to the PTS
- introduction of the `builddadm` unix group with access to all `buildds` for investigations
- introduction of a YAML-based wanna-build API for passing on more information to the `buildds`
- <https://buildd.debian.org/status/> is now maintained by Mehdi Dogguy and serving live data
- the status interface is now also the recommended interface, available by default from the “redesigned” front page

Last year's "What's next?" slide

- ✓ move away from raff.debian.org to another machine.
- ✓ integration of the unofficial autobuilders into the **debian.org** setup (i.e. *volatile*, *experimental*)
- ✓ putting buildds until DSA's control (mainly arm, s390)
- ✓ further bringing the software in sync
- ✓ pull wanna-build onto a sane database

What's next?

- finishing non-free autobuilding
- adding a capability to trigger archive rebuilds in unstable against selected packages from experimental (almost ready)

Who's currently doing wanna-build/buildd integration work?

- Andreas Barth, Kurt Roeckx, Marc Brockschmidt and me
- **also involved:** Roger Leigh (sbuid maintainer), Mehdi Dogguy (web pages)

Current set of builddd maintainers

- **amd64**: Kurt Roeckx, Frederik Schüler
- **alpha**: Kurt Roeckx
- **arm**: Aurelien Jarno
- **armel**: Riku Voipio
- **hppa**: dann frazier
- **i386**: Kurt Roeckx, Philipp Kern
- **ia64**: LaMont Jones, Andreas Barth
- **mips{,el}**: Peter De Schrijver, Andreas Barth
- **powerpc**: Wouter Verhelst, Philipp Kern
- **s390**: Bastian Blank, Andreas Barth
- **sparc**: Cyril Brulebois
- **kfreebsd-{i386,amd64}**: Cyril Brulebois, Aurelien Jarno
- mostly documented in #debian-builddd's channel topic

Current count of builddds

amd64:	2	arm:	4 ext
alpha:	2	armel:	7 d.o, 1 ext
hppa:	3	s390:	1 d.o, 2 ext
i386:	2		
ia64:	2		
mips:	3		
mipsel:	4		
powerpc:	2		
sparc:	3		
kfreebsd-i386:	2		
kfreebsd-amd64:	2		

Current count of builddds

amd64:	2
alpha:	2
hppa:	3
i386:	2
ia64:	2
mips:	3
mipsel:	4
powerpc:	2
sparc:	3
kfreebsd-i386:	2
kfreebsd-amd64:	2

arm: 4 ext

armel: 7 d.o, 1 ext

s390: 1 d.o, 2 ext

⇒ Most are now part of the
Debian infrastructure

Current count of builddds

amd64:	2
alpha:	2
hppa:	3
i386:	2
ia64:	2
mips:	3
mipsel:	4
powerpc:	2
sparc:	3
kfreebsd-i386:	2
kfreebsd-amd64:	2

arm: 4 ext

armel: 7 d.o, 1 ext

s390: 1 d.o, 2 ext

⇒ Most are now part of the
Debian infrastructure

... setting one up is as easy as
getting DSA to run puppet and
to follow the step-by-step guide

Questions? Comments?

- What would **you** need from the autobuilding system?
- What **issues** do you have with the autobuilders?
- Our email address: `debian-wb-team@lists.debian.org`