Open is not enough: benefits from Debian as an integrated, community-driven computing platform

Yaroslav O. Halchenko & Michael Hanke

Debian Project
Dartmouth College, New Hampshire
University of Magdeburg, Germany

SEA 2013, Boulder CO
March 2 2013
Can someone point me towards info. on how one can duplicate the packages installed on one Debian system on another? Going through dselect by hand seems just a bit too tedious for multiple installations... Thanks.

--

Tres Hofmeister
tres@ncar.ucar.edu

Research Applications Program
National Center for Atmospheric Research

Open is not enough. Let’s take the next step: an integrated, community-driven computing platform for neuroscience

Yaroslav O. Halchenko\textsuperscript{1,2,3,*} and Michael Hanke\textsuperscript{4,5,3,*}

1 Center for Cognitive Neuroscience, Dartmouth College, Hanover, NH, USA
2 Department of Psychological and Brain Sciences, Dartmouth College, Hanover, NH, USA
3 Debian Project, http://www.debian.org
4 Department of Experimental Psychology, Otto-von-Guericke University, Magdeburg, Germany
5 Center for Behavioral Brain Sciences, Magdeburg, Germany

The last 5 years have seen dramatic improvements in the collaborative research infrastructure. A need for open research tools has been identified (Ince et al., 2012), and one solution has been clearing houses, such as the INCF Software Center\textsuperscript{1}, and the NITRC\textsuperscript{2} portal, which facilitate efforts of
Who is talking?

I and
Who is talking?

http://www.pymvpa.org
http://www.debian.org
http://neuro.debian.net
Poline et al., Data sharing in neuroimaging research, 10.3389/fninf.2012.00009
Our belief #1

There is no alternative to free and open-source software for scientific research.
Common problem: Availability != Accessibility

To download and install A on your system, you need to:
- Register/Request an account from the B
- Read these notes for C and D systems
- Download the source code and/or appropriate binaries from E
- Remove old versions
- Install software
- Test software
Common problem: Availability $\neq$ Accessibility

"Standard" model of scientific software deployment

To download and install A on your system, you need to:

- Register/Request an account from the B
- Read these notes for C and D systems
- Download the source code and/or appropriate binaries for your system from E
- Remove old versions
- Install software
- Test software
Common problem: Availability != Accessibility

“Standard” model of scientific software deployment

To download and install A on your system, you need to:

- Register/Request an account from the B
- Read these notes for C and D systems
- Download the source code and/or appropriate binaries for your system from E
- Remove old versions
- Install software
- Test software
- If you decide to go forth with building from source code, good luck!
Possible user experiences

Not so positive

- Software developers have no access to my system X and I need to build a few (dozens) of additional pre-requisites first:
  - I better ask our IT personnel to deploy it – usually takes less than a month
  - PhD students have all the time in the universe . . .
Possible user experiences

Not so positive

- Software developers have no access to my system X and I need to build a few (dozens) of additional pre-requisites first:
  - I better ask our IT personnel to deploy it – usually takes less than a month
  - PhD students have all the time in the universe . . .

Positive

- Only an hour and “A” seems to be running – I can get back to do research again!
Possible user experiences

Not so positive

- Software developers have no access to my system X and I need to build a few (dozens) of additional pre-requisites first:
  - I better ask our IT personnel to deploy it – usually takes less than a month
  - PhD students have all the time in the universe . . .

Positive

- Only an hour and “A” seems to be running – I can get back to do research again!
- Student’s life is short:
  - Having A on one box should be enough
  - I better not have to upgrade it ever again
  - I will not even think about F and G alternatives
  - I value others’ time: let me blog a “HOWTO install A v. x.y.z on C”
Hanke, M. (2012). Share your tools! But fear the wombat!

http://youtu.be/8t6znEOEDVo

Not so positive

- Software developers have no access to my system X and need to build a few (dozens) of additional pre-requisites first:
  - I better ask our IT personnel to deploy it – usually takes less than a month
  - PhD students have all the time in the universe . . .

Positive

- Only an hour and A seems to be running – I can get back to do research again!
- Student’s life is short:
  - Having A on one box should be enough
  - I better not have to upgrade it ever again
  - I will not even think about F and G alternatives
  - I value others’ time: let me blog a “HOWTO install A v. x.y.z on C”
Scientists report:

GNU/Linux users report the lowest average time they need to invest in maintenance of their personal computing environment (5.77 h/month).

GNU/Linux’es

http://futurist.se/gldt
GNU/Linux'es

http://futurist.se/gldt

A. Lundqvist, D. Rodic - futurist.se/gldt
Published under the GNU Free Documentation License

http://futurist.se/gldt

GNU/Linux Distribution Timeline
Version 12.2
A. Lundqvist, D. Rodic - futurist.se/gldt
Published under the GNU Free Documentation License
Our belief #2

Debian provides the best proven, scalable, and sustainable approach toward free and open-source software platform for scientific research.
Debian’s recipe

- Open standards
- Vibrant community: democracy and do-o-cracy
- Integration starting with the core OS
- Binary distribution based on source packages
Debian’s recipe: Binary distribution

- Open standards
- Vibrant community: democracy and do-o-cracy
- Integration starting with the core OS
- Binary distribution based on source packages

Extensible distribution and archival infrastructure

http://xkcd.com/303
Debian’s recipe

- Open standards
- Vibrant community: democracy and do-o-cracy
- Integration starting with the core OS
- Binary distribution based on source packages
- Extensible distribution and archival infrastructure
Legal assurance

Was: Register/Request an account from the B

3 Debian archive areas

- main – software under DFSG compliant licenses
- contrib – DFSG-wannabees (depend on non-free) or helpers (e.g., matlab-support)
- non-free – (re-)distributable but restricted

---

1See Debian Social Contract http://www.debian.org/social_contract
Legal assurance Was: Register/Request an account from the B

3 Debian archive areas

- main – software under DFSG compliant licenses
- contrib – DFSG-wannabees (depend on non-free) or helpers (e.g., matlab-support)
- non-free – (re-)distributable but restricted

Pros

- you know that everything is free and open-source
- eases integration
- helps to assure longevity
- http://popcon.debian.org is there to replace “registration” pages

---

1 See Debian Social Contract http://www.debian.org/social_contract
Legal assurance

Was: Register/Request an account from the B
Debian archive
areas
- main – software under DFSG compliant licenses
- contrib – DFSG-wannabees (depend on non-free) or helpers (e.g.,
  matlab-support)
- non-free – (re-)distributable but restricted

Pros
- you know that everything is free and open-source
- eases integration
- helps to assure longevity

http://popcon.debian.org is there to replace "registration" pages

See Debian Social Contract
http://www.debian.org/social_contract
Legal assurance  

Was: Register/Request an account from the B

Recommendations

- use standard OSI (http://opensource.org) and DFSG compliant licenses
- do not restrict the domain of application without need
- do not insert code/data with questionable license terms
- maintain a top-level list of licenses used in your project
- beware-of and/or state your trademark policies

Open is not enough
Building: developers

Image by Lewis Hine, 1930
Building: well-engineered software

Image by Lewis Hine, 1930
Image by Ad Meskens
Building: “state-of-art”
Building Was: If you decide to go with building from source code, good luck!

- Debian packaging strictly follows Debian Policy\(^1\)
- 3rd-party modules become 1st-class citizens
- Figure out dependencies once – specify in debian/control
- Figure out building/installation once – specify in debian/rules

\(^1\)http://www.debian.org/doc/debian-policy/
Building Was: If you decide to go with building from source code, good luck!

- Debian packaging strictly follows Debian Policy\(^1\)
- 3rd-party modules become 1st-class citizens
- Figure out dependencies once – specify in debian/control
- Figure out building/installation once – specify in debian/rules

Pros

- anyone can build any PACKAGE: apt-get source -b PACKAGE
- simplifies contributions
- allow for archive-wide QA rebuilds
  - library migrations
  - new compilers
  - new ports

\(^1\)http://www.debian.org/doc/debian-policy/
Building Was: If you decide to go with building from source code, good luck!

**Recommendations**

- keep source distribution modular:
  - code-vs-data
  - keep 3rd party as 3rd party – do not clone
- have a deterministic version
- provide exhaustive specification of (build-)dependencies
- assure API/ABI compatibility if you deliver libraries:
  - [http://upstream-tracker.org/](http://upstream-tracker.org/)
  - [http://github.com/lvc/abi-compliance-checker](http://github.com/lvc/abi-compliance-checker)
Building

Was: If you decide to go with building from source code, good luck!

Recommendations

- keep source distribution modular:
  - code-vs-data
  - keep 3rd party as 3rd party – do not clone
- have a deterministic version
- provide exhaustive specification of (build-)dependencies
- assure API/ABI compatibility if you deliver libraries:
  - http://upstream-tracker.org/
  - http://github.com/lvc/abi-compliance-checker

Above recommendations will help to deliver your software to any integration/distribution platform, not just Debian.
Testing

Geeks and repetitive tasks

- Time spent
- Task size

<table>
<thead>
<tr>
<th>Geek</th>
<th>Non-Geek</th>
</tr>
</thead>
<tbody>
<tr>
<td>- does it manually</td>
<td></td>
</tr>
<tr>
<td>- writes script to automate</td>
<td></td>
</tr>
<tr>
<td>- runs script</td>
<td></td>
</tr>
</tbody>
</table>

- Gets annoyed |
- Loses |
- Wins |
- Makes fun of geek's complicated method
Testing

Was: Test software

- debian/rules: can exercise tests while building the binary packages
Testing  

Was: Test software

- debian/rules: can exercise tests while building the binary packages

Pros

- obtain a CI farm with $> 13$ architectures/ports
- have I mentioned archive wide QA rebuilds?  
  **Go ahead of the OS not behind**
- preclude problems on users’ deployments
Testing

**Was:** Test software

- debian/rules: can exercise tests while building the binary packages

**Pros**

- obtain a CI farm with > 13 architectures/ports
- have I mentioned archive wide QA rebuilds? **Go ahead of the OS not behind**
- preclude problems on users’ deployments

- autopkgtest: automatic as-installed package testing

**Pros**

- exercise testing on installed machines
- could be “heavy” tests
Testing  

**Was:** Test software

---

**Recommendations**

- share your unit-, regression-, integration- tests
- work on your tests coverage
Testing  Was: Test software

Recommendations

- share your unit-, regression-, integration- tests
- work on your tests coverage
- more problems we catch before delivering to users – less of WOMBAT
Installation, upgrades  

Was: Remove old versions; Install software

Install simple editor

```
apt-get install gedit
```

Install workload management system

```
apt-get install condor
```

Install the entire suite of packages for meteorology

```
apt-get install science-meteorology
```

Keep the whole system up-to-date

```
apt-get update && apt-get upgrade
```
Installation, upgrades

Was: Remove old versions; Install software

Install simple editor

```
apt-get install gedit
```

Install workload management system

```
apt-get install condor
```
Installation, upgrades

Was: Remove old versions; Install software

Install simple editor

```
apt-get install gedit
```

Install workload management system

```
apt-get install condor
```

Install the entire suite of packages for meteorology

```
apt-get install science-meteorology
```
Installation, upgrades

Was: Remove old versions; Install software

Install simple editor

`apt-get install gedit`

Install workload management system

`apt-get install condor`

Install the entire suite of packages for meteorology

`apt-get install science-meteorology`

Keep the whole system up-to-date

`apt-get update && apt-get upgrade`
Climate/atmospheric research FOSS in Debian


**Task pages:** astronomy, astronomy-dev, bci, biology, chemistry, dataacquisition, dataacquisition-dev, distributedcomputing, electronics, electrophysiology, engineering, engineering-dev, geography, highenergy-physics, highenergy-physics-dev, imageanalysis, linguistics, machine-learning, mathematics, mathematics-dev, meteorology, meteorology-dev, nanoscale-physics, nanoscale-physics-dev, neuroscience-cognitive, neuroscience-datasets, neuroscience-modeling, numericalcomputation, physics, physics-dev, psychophysics, robotics, simulations, statistics, typesetting, viewing
# Debian Science Meteorology packages

## Official Debian packages with high relevance

**Aweather**

*Advanced Weather Monitoring Program*

http://lug.rose-hulman.edu/proj/aweather

Maintainer: Debian Science Maintainers (Andy Spencer)

<table>
<thead>
<tr>
<th>Popcon: 16 users (5 upd.)*</th>
<th>License: DFSG free Official Debian package Git</th>
</tr>
</thead>
<tbody>
<tr>
<td>Versions and Archs</td>
<td>Edit Debtags</td>
</tr>
</tbody>
</table>

AWeather is an advanced weather program which is designed to be used by weather enthusiasts. AWeather is not a weather dockapp that simply displays a pre-computed forecast. It is designed to be an easy-to-use program that integrates a variety of weather data in a simple unified interface. AWeather currently supports radar and weather alerts from the United States National Weather Service.

**Cdo**

*Climate Data Operators*

https://code.zmaw.de/projects/cdo

Maintainer: Alastair McKinstry

<table>
<thead>
<tr>
<th>Popcon: 31 users (3 upd.*)</th>
<th>License: DFSG free Official Debian package</th>
</tr>
</thead>
<tbody>
<tr>
<td>Versions and Archs</td>
<td>Edit Debtags</td>
</tr>
</tbody>
</table>

Climate Data Operators are a collection of command line Operators to manipulate and analyse Climate model Data. Supported data formats are GRIB, netCDF, SERVICE, EXTRA and IEG. There are more than 400 operators available.

**Cmip5-cmor-tables**

*CMIP5 tables for the Climate Model Output Rewriter library*

http://www2-pcmdi.llnl.gov/cmor

Maintainer: Alastair McKinstry

<table>
<thead>
<tr>
<th>Popcon: 0 users (0 upd.)*</th>
<th>License: DFSG free Official Debian package Git</th>
</tr>
</thead>
<tbody>
<tr>
<td>Versions and Archs</td>
<td>Go tagging</td>
</tr>
</tbody>
</table>
Climate/atmospheric research FOSS in Debian

Debian Science:
http://wiki.debian.org/DebianScience
http://blends.alioth.debian.org/science/tasks/meteorology

Alastair McKinstry
mckinstry@debian.org

Ireland’s High-Performance Computing Centre
http://www.ichec.ie

Debian GIS:
http://wiki.debian.org/DebianGis
http://blends.alioth.debian.org/gis/tasks/

Francesco P. Lovergine
frankie@debian.org

Open is not enough
Climate/atmospheric research FOSS in Debian

Ireland's High-Performance Computing Centre
http://www.ichec.ie

Debian GIS:
http://wiki.debian.org/DebianGis
http://blends.alioth.debian.org/gis/tasks/

Francesco P. Lovergine:
frankie@debian.org

Open is not enough
Climate/atmospheric research FOSS in Debian

Debian GIS:
http://wiki.debian.org/DebianGis
http://blends.alioth.debian.org/gis/tasks/

Francesco P. Lovergine:
<frankie@debian.org>

Debian Science:
http://wiki.debian.org/DebianScience
http://blends.alioth.debian.org/science/tasks/meteorology

Alastair McKinstry:
<mckinstry@debian.org>

Ireland’s High-Performance Computing Centre:
http://www.ichec.ie

Open is not enough
Debian Science : http://wiki.debian.org/DebianScience
http://blends.alioth.debian.org/science/tasks/meteorology
Alastair McKinstry <mckinstry@debian.org>
Ireland’s High-Performance Computing Centre http://www.ichec.ie
Debian Science : http://wiki.debian.org/DebianScience
    http://blends.alioth.debian.org/science/tasks/meteorology
Alastair McKinstry <mckinstry@debian.org>
Ireland’s High-Performance Computing Centre http://www.ichec.ie

Debian GIS : http://wiki.debian.org/DebianGis
    http://blends.alioth.debian.org/gis/tasks/
Francesco P. Lovergine” <frankie@debian.org>
A Debian Pure Blend is a Debian internal project which assembles a set of packages that might help users to solve certain tasks of their work. The list on the right shows the tasks of Debian GIS.

Tasks page

This is a list of the Tasks Debian GIS is made of:

- **Data - Debian GIS data**
  This metapackage will install some packages providing data that can be used by different GIS applications.

- **Gps - GPS related programs**
  Set of Debian packages which are dealing with GPS devices and data.

- **Openstreetmap - OpenStreetMap related programs**
  Set of Debian packages which are dealing with OpenStreetMap data.

- **Remote sensing - Remote sensing and earth observation**
  Debian packages which are dealing with Remote Sensing (for instance Synthetic Aperture Radar (SAR) processing (interferometry, polarimetry, data visualization, etc) and earth observation.

- **Statistics - Statistics with geographical data**
  Set of Debian packages which are useful for doing statistics with geographical data.

- **Map server - Present geographic information via web map server**
  Debian packages which are dealing with geographical information to be presented for the web on so called map tile servers. These are pretty useful when trying to setup an OpenStreetMap tile server but not restricted to OpenStreetMap data only.

- **Workstation - Geographic Information Systems (GIS) workstation**
  This task sets up your system to be a GIS workstation to process geographical information and make maps.
Additional benefits (in examples)

- Longevity
- Reproducibility
Longevity

XGKS: Graphical Kernel System for the X Window System
http://xgks.sourceforge.net

- developed within unidata.ucar.edu
- now considered legacy:
  2.6 in 2000, 2.6.2 release in 2005-02-03
- still used by
  - Ferret: http://www.ferret.noaa.gov
  - UV-CDAT: http://uvcdat.llnl.gov from LLNL, DOE
- and is still maintained in Debian
Longevity

Recommendations

- do not under-estimate the number of your users
- prepare your software for the after-life:
  - clean legal terms
  - commented code
  - reliable build infrastructure
Reproducibility

Reproducibility

  
  **Example**: state of a development box in 2003:

  ```
  debootstrap --arch=i386 --include=build-essential,emacs20,netcdf3 \ 
  potato /tmp/potato http://archive.debian.org/debian
  ```

- [http://snapshots.debian.org](http://snapshots.debian.org) – snapshots of the archive

  ```
  debootstrap --arch=i386 --include=build-essential,libnetcdf-dev \ 
  testing /tmp/testing \ 
  http://snapshot.debian.org/archive/debian/20080407T000000Z/
  ```
Reproducibility

  
  **Example:** state of a development box in 2003:

  ```
  debootstrap --arch=i386 --include=build-essential,emacs20,netcdf3 \ 
  potato /tmp/potato http://archive.debian.org/debian
  ```

- `schroot` – enter into the environment upon your convenience

  ```
  > cat /etc/schroot/chroot.d/lenny-20080407
  [potato]
  description=Antique Debian from 2003
  directory=/tmp/potato
  users=YOURLOGIN
  > schroot -c potato
  ```
Reproducibility

  
  Example: state of a development box in 2003:

```
> cat /etc/schroot/chroot.d/lenny-20080407
[potato]
  description=Antique Debian from 2003
  directory=/tmp/potato
  users=YOURLOGIN
> schroot -c potato
```

- schroot – enter into the environment upon your convenience

```
debootstrap --arch=i386 --include=build-essential,emacs20,netcdf3 \
  potato /tmp/potato http://archive.debian.org/debian
```

- http://snapshots.debian.org – snapshots of the archive

```
debootstrap --arch=i386 --include build-essential,libnetcdf-dev \
  testing /tmp/testing \
  http://snapshot.debian.org/archive/debian/20080407T000000Z/
```
Take home

If you care about open, accessible and reproducible science:

Get all software you care about into Debian!
Thanks

http://www.debian.org
http://neuro.debian.net

Yaroslav O. Halchenko
yoh@debian.org
Michael Hanke
mih@debian.org

about the slides:
available at
http://neuro.debian.net/#publications

copyright © 2013
Yaroslav O. Halchenko & Michael Hanke, front/end slide style inspired by Stefano Zacchioli
CC BY-SA 3.0 — Creative Commons Attribution-ShareAlike 3.0
apt-get install -t audience [-a microphone] questions
Debian, 19 years old

- ≈ 18’704 (main), 222 (contrib), 506 (non-free) source packages
- ≈ 38’537 (main), 141 (contrib), 289 (non-free) binary packages
- largest n. of ports among mainstream distros (11 official, 11 unofficial)
  - 2 non-Linux ports: GNU/kFreeBSD + (unofficial Hurd)
Debian, 19 years old

- \( \approx 18'704 \) (main), 222 (contrib), 506 (non-free) source packages
- \( \approx 38'537 \) (main), 141 (contrib), 289 (non-free) binary packages
- largest n. of ports among mainstream distros (11 official, 11 unofficial)
  - 2 non-Linux ports: GNU/kFreeBSD + (unofficial Hurd)
- 12 stable releases
  - The latest stable release: 6.0 Squeeze, February 6th 2011
  - Upcoming stable release: 7.0 Wheezy, TBA 2013
- Released \( \approx 2-3 \) years
- Security support for 3 years
- Upgradable
Debian, 19 years old

- ≈ 18'704 (main), 222 (contrib), 506 (non-free) source packages
- ≈ 38'537 (main), 141 (contrib), 289 (non-free) binary packages
- largest n. of ports among mainstream distros (11 official, 11 unofficial)
  - 2 non-Linux ports: GNU/kFreeBSD + (unofficial Hurd)
- 12 stable releases
- 2 1/2 rolling releases: testing, unstable (sid), experimental
Debian, 19 years old: > 100 derivatives

- ≈ 18’704 (main), 222 (contrib), 506 (non-free) source packages
- ≈ 38’537 (main), 141 (contrib), 289 (non-free) binary packages
- largest n. of ports among mainstream distros (11 official, 11 unofficial)
  - 2 non-Linux ports: GNU/kFreeBSD + (unofficial Hurd)
- 12 stable releases
- 2 1/2 rolling releases: testing, unstable (sid), experimental
- > 130 “derivative” distributions (source: distrowatch.com)
Debian, 19 years old

- ≈ 18’704 (main), 222 (contrib), 506 (non-free) source packages
- ≈ 38’537 (main), 141 (contrib), 289 (non-free) binary packages
- largest n. of ports among mainstream distros (11 official, 11 unofficial)
  - 2 non-Linux ports: GNU/kFreeBSD + (unofficial Hurd)
- 12 stable releases
- 2 1/2 rolling releases: testing, unstable (sid), experimental
- > 130 “derivative” distributions (source: distrowatch.com)
- ≈ 900 active DDs + ≈200 DMs + thousands other contributors
Debian: the Universal OS

- ≈ 18’704 (main), 222 (contrib), 506 (non-free) source packages
- ≈ 38’537 (main), 141 (contrib), 289 (non-free) binary packages
- largest n. of ports among mainstream distros (11 official, 11 unofficial)
  - 2 non-Linux ports: GNU/kFreeBSD + (unofficial Hurd)
- 12 stable releases
- 2 1/2 rolling releases: testing, unstable (sid), experimental
- > 130 “derivative” distributions (source: distrowatch.com)
- ≈ 900 active DDs + ≈200 DMs + thousands other contributors

Success.
Debian GNU/Linux release timeline

- squeeze
- lenny
- etch
- sarge
- woody
- potato
- slink
- hamm

Release versions:
- squeeze (6.0)
- lenny (5.0)
- etch
- sarge
- woody
- potato
- slink
- hamm

Timeline:
- 1997 to 2013

 Updates:
- 2012/06/22

(Debian/Dartmouth + Magdeburg)

Open is not enough

SEA 2013
NeuroDebian VM in VirtualBox

http://neuro.debian.net/vm.html

Open is not enough
Ways to contribute

http://wiki.debian.org/ProjectNews/HowToContribute
http://raphaelhertzog.com/2011/06/30

- reportbug (+ patches)
- Internationalization (i18n):
  http://www.debian.org/doc/manuals/intro-i18n
- packaging
  - Luca’s tutorial
    `apt-get install packaging-tutorial`
    http://www.lucas-nussbaum.net/blog/?p=640
  - Bootstrap packaging of Python modules:
    `py2dsc` (python-stddeb package)
- Good night reading: Debian Policy
- Seek mentor/sponsor-ship: http://mentor.debian.org
- Become “Debian Maintainer”:
  http://wiki.debian.org/DebianMaintainer
- Become “Debian Developer”:
  http://wiki.debian.org/DebianDeveloper
And it seems to work...

Debian has found the sweet balance between agile behavior and formal processes that make it possible to keep up with the rapid innovation of the upstream packages, while still maintaining structure and organization.

–Luis Ibanez, Kitware.com (makers of VTK, ITK, . . . )

prevailing of GNU/Linux systems as a computing platform in neuroscience. Moreover, a concern was raised that virtualization is not a feasible solution to overcome limitations of any particular platform or to provide a convenient multiprocessor working environment. We were surprised by these comments, because they are in contrast to what we experience daily while working with software developers worldwide to integrate neuroscience software into the NeuroDebian project.
Debian GNU/Linux 6.0 is the first GNU/Linux distribution release ever to offer comprehensive support for magnetic resonance imaging (MRI) based neuroimaging research.

–Release notes Debian 6.0 (squeeze)

And it seems to work...

Moreover, a concern was raised that virtualization is not a feasible solution to overcome limitations of any particular platform or to provide a convenient multiplatform working environment. We were surprised by these comments, because they are in contrast to what we experience daily while working with software developers worldwide to integrate neuroscience software into the NeuroDebian project.

What is NeuroDebian?

**QA/Testing**
- Build-time tests
- Maintainer tests
- Interoperability testing

**Virtual machine**
- Compatibility
- Reproducable research
- Portability
- Teaching

**Packaging**
- Mentoring
- Maintenance
- Legal checks
- Modularisation

**Communication**
- Online
  - Mailing lists
  - IRC (#neurodebian on OFTC)
  - identi.ca/twitter (neurodebian)
  - NeuroDebian Insider Blog
  - team@neuro.debian.net
- Personal
  - Talks
  - Conference booth
  - Stalking

**Repository**
- Software
  - Debian
  - i386
  - amd64
  - Ubuntu
  - Library/compiler transitions
- Data
  - Usage statistics
  - Snapshoting
  - DE
  - GR
  - US-CA
  - US-NH
  - US-TN

**Debian**
- Archive
  - BTS
  - QA
  - Blends
  - Debian Science
  - Debian Med
  - i386
  - amd64
  - arme
  - mips(el)
  - FreeBSD-i386/amd64
  - kfreebsd-i386/amd64
  - ...
  - Derivatives

**Open is not enough**

H² (Debian/Dartmouth+Magdeburg)
But, but my true love is a fruit and I’m married to monster!

- NeuroDebian virtual machine (32/64bit, multi-core)
- Most convenient solution for Mac OS X, Windows
- Base image with setup wizzard, fully functional within minutes
- Great for teaching, workshops, development, analysis

http://neuro.debian.net/vm.html
How does software benefit from Debian?

- Extended reach
  - one stable release, two rolling “release” flavors
  - $\approx 130$ derivative distributions (distrowatch.org)
How does software benefit from Debian?

**Debian "unstable"**
- Release: never
- Updates: multiple times per day
- Security support: none (implicit)

**Debian "testing"**
- Release: never/continuously
- Updates: daily
- Security support: yes

**Debian "stable"**
- Release: when ready
- Updates: every two months (only critical fixes)
- Security support: one year after next stable (≈ 3 years)

- e.g. Ubuntu
- e.g. Linux Mint
- e.g. MEPIS

Release: yes
Updates: multiple times per day
Security support: none (implicit)
How does software benefit from Debian?

- Extended reach
  - one stable release, two rolling “release” flavors
  - \( \approx 130 \) derivative distributions (distrowatch.org)

- Mutual awareness
  - Explicitly documented dependencies
  - Synchronized transitions

- Less maintenance work through modularity
  - 3rd-party software in dedicated packages maintained by someone else

- Continuous integration testing
  - 13 hardware architectures
  - Three kernels
  - Continuous automated testing for
    - Build success
    - Clean installation/de-installation, Availability of dependencies
    - Policy compliance
    - Package conflicts
But I only care about Ubuntu!

No, you don’t!

- Most software we care about comes (almost) 1:1 from Debian (SciPy, VTK, ITK, ...)
- No LTS for neuroscience (NumPy only since 10.04)

Go Debian!

- Developers: Get it right in Debian, have it work in Debian/Ubuntu/Mint/aptosid/Mepis/... (at no additional cost)
- Users: Stable release with 3-4 years support for all software
- Scientists: Want your research tool to be found and used? Include it in the largest software archive in the world.
NeuroDebian world-map

NeuroDebian Usage Statistics
Data from March 2009 to April 2012 (NeuroDebian repository only) aggregated per city or region as defined by GeoIP
Package downloads
>8192 >2048 >512 >128 >32
Number of unique IPs
>2048 >512 >128 >32 >8

NeuroDebian Repository Mirror