Proceedings of the 11th Debian Conference

DebConf10 Proceedings Team

August 1th–7th, 2010

Preface

DebConf, the annual Debian Developers Meeting, is an an open event aimed to improve communication between everyone involved in Debian development and to reach out to a wider community interested in Free Software and related issues. In addition to a full schedule of technical, social and policy talks, DebConf provides an opportunity for developers, contributors and other interested people to meet in person and work together more closely. It has taken place annually since 2000 in locations as varied as Argentina, Canada, Finland and Mexico. Debian Conferences have featured speakers from around the world. They have also been extremely beneficial for developing key Debian software components, including the new Debian Installer, and for improving Debian's internationalization.

DebConf10, the 11th annual DebConf was held on the Manhattan campus of Columbia University from August 1 to August 7 2010. These proceedings contain abstracts or brief descriptions of the technical and social events held during DebConf10.

More information about DebConf10 and the Debian Developers Meeting can be found on the DebConf website at `http://www.debconf.org/`.

This document has been compiled by the DebConf10 Proceedings Team, on behalf of the DebConf10 Organization Team.

Contents

Chapter 1

Abstracts

1.1 General

1.1.1 Batch Queuing Systems BoF

Michael Banck ⟨mbanck@debian.org⟩

There are a couple of batch queuing systems in Debian or as prospective packages - Gridengine, Torque and others. However, it seems the number of people willing to maintain them is quite small among all of them. This BoF will try to get the present maintainers/prospective maintainers of batch queuing in one room and discuss the way forward - e.g. should we pick one queuing system and concentrate efforts for it? Does it make sense to standardize on some policies among them?

1.1.2 Bits from the DPL

Stefano Zacchiroli ⟨zack@debian.org⟩

Bits from the DPL, you know the drill :)

Some topics that will addressed: - how are we? - are we aging properly? - what's our role in the (always changing) ecosystem of FLOSS distributions? - are we doing well in (continuously) attracting new developers? - are we doing well in communicating within the project and with the external world? - how's doing the (newbie) DPL? - how are we wrt to ... ? - total world domination is *just* around the corner

1.1.3   Bits from the Release Team

Mehdi Dogguy ⟨dogguy@pps.jussieu.fr⟩, Philipp Kern ⟨pkern@debian.org⟩,
Adam D. Barratt ⟨adsb@debian.org⟩

1.1.4   BoF Debian Publicity

Jeremiah C. Foster ⟨jeremiah@jeremiahfoster.com⟩

1.1.5   Bugs, the BOF

Don Armstrong ⟨don@debian.org⟩

Open discussion about possible improvements and current issues with http://bugs.debian.org

1.1.6   Classical Keysigning Party

Aníbal Monsalve Salazar ⟨anibal@debian.org⟩

We will do the Classical Keysigning ritual at this event.

1.1.7   Cross-distro dependency resolution: reusing solvers among distros

Stefano Zacchiroli ⟨zack@debian.org⟩

All distributions use some kind of meta-data describing relationships between packages such as Dependencies, Conflicts, Virtual Packages, etc. Existing package managers are often specific to a packaging format, or even use implicit assumptions on the meaning of package relations in the context of a distribution.

This talk addresses the issue of how we can overcome this seclusion by resolving inter-package relations in a way that is independent from a specific distribution.

We will present CUDF (Common Upgradeability Description Format) which permits to encode upgrade scenarios coming from different packaging systems, including in particular .deb-based and .rpm-based distributions. The format comes with a clear semantics which makes it possible to share resolution tools among distributions, and to adapt general-purpose solvers to the needs of FLOSS distributions.

We also show how, in the context of the Mancoosi project, we are using the CUDF format to improve the solving abilities of state-of-the-art package managers in distributions such as Debian, Mandriva, and Caixa Magica.

1.1.8 Day Trip to Coney Island

Clint Adams ⟨schizo@debian.org⟩

We will all be going to Coney Island for fun and adventure. Explore historic Brighton Beach! Ride the Cyclone rollercoaster! Swim in the ocean! See the circus sideshow! Enjoy a baseball game! Fly kites! Eat food! Play Frisbee!

Or do none of the above and just hang out with your friends outside and have a good time.

1.1.9 DebConf11 in Bosnia

Adnan Hodzic ⟨adnan@foolcontrol.org⟩

1.1.10 Debbugs

Don Armstrong ⟨don@debian.org⟩

Debbugs is the software that underlies bugs.debian.org; currently has over 580,000 bugs filed in it, over 480,000 of which have been fixed. Currently in use by thousands of Debian developers and maintainers and unknown numbers of Debian users daily, it is continuously undergoing upgrades to better service the needs of Developers.

1.1.11 Debconf closing plenary

Stefano Zacchiroli ⟨zack@debian.org⟩, Adnan Hodzic ⟨adnan@foolcontrol.org⟩, Micah Anderson ⟨micah@debian.org⟩

The closing plenary would be the final wrap-up from Debconf. It will be the companion book-end to the opening plenary at the beginning. It will be the general good-bye, providing some closure, final words, and report-backs. Ideally this closing plenary would be scheduled as the very last thing, and would not be in competition with other things. It would likely be brief, and include things from a number of contributors (zack for BSP/RCBC prizes; dkg coordinating talk report-backs; adnan for bosnia see you next time; etc.)

1.1.12 Debconf opening plenary

Gabriella Coleman ⟨biella@gmail.com⟩

The opening plenary would be the first thing that would be scheduled to kick-off Debconf (after debcamp). It would be the general welcoming to the conference, providing a opening cohesion and hitting a few key things. Ideally the plenary would be scheduled first thing, and not in competition with other things. It would be brief (30 minutes probably).

1.1.13 Debian Day opening Plenary

Gabriella Coleman ⟨biella@gmail.com⟩, Hans-Christoph Steiner ⟨hans@at.or.at⟩

Welcoming people to Debian Day

1.1.14 Debian GNU/Hurd

Emilio Pozuelo Monfort ⟨pochu@debian.org⟩, Michael Banck ⟨mbanck@debian.org⟩

This talk will present the status of the GNU/Hurd OS and of the Debian port, its future, and how you can help it.

1.1.15 Debian IRC Operators BoF

Michael Banck ⟨mbanck@debian.org⟩

Discussion about #debian (and maybe #debian-devel) channel operation. What to do about non-english channels, what should be considered on- and off-topic, and how strict to enforce it, etc.

1.1.16 Debian Policy BoF

Russ Allbery ⟨rra@debian.org⟩

General discussion of the current state and workflow for Debian Policy, possible future changes to the document structure and workflow, and general discussion of how to get more people involved and how to help Policy evolve at the pace desired by the project.

1.1.17 Debian Wiki BoF

Franklin Piat ⟨fpiat@klabs.be⟩

You want a better wiki? You have some ideas? You want a feature... Come, Discuss, Contribute!

1.1.18   Debian for Shy People

Asheesh Laroia ⟨asheesh@asheesh.org⟩

Debconf attendees love Debian. For some, our pride in the project combines with self-confidence to become a sense of empowerment. For others, we fear we are not quite worthy.

Let's discuss how prospective contributors feel welcomed or scared-off by the new maintainer process and debian-mentors, how existing contributors see Debian, and what we can do to help technically-sharp but socially-shy people give Debian their best.

1.1.19   Debian/kFreeBSD from the FreeBSD Point of View

Philip Paeps ⟨philip@freebsd.org⟩

This talk briefly introduces the history and the goals of the FreeBSD Project and its development model.

For most of the history of the FreeBSD Project, FreeBSD was "the" distribution. In recent years, a number of people have taken on the task of taking the FreeBSD operating system, or in the case of Debian only the kernel, and making it part of a different project. The FreeBSD Project encourages these endeavors and is pleased to see exposure to new environments.

Debian/kFreeBSD is unique in the FreeBSD distribution "ecosystem" in that it takes only the kernel and replaces the entire userspace, this poses some interesting new problems that we have not seen in the past but also brings new opportunities.

1.1.20   FOSSology and Debian

Matt Taggart ⟨taggart@debian.org⟩

The FOSSology Project is a Free Open Source Software (FOSS) project built around an open and modular architecture for analyzing software. Existing modules include license and copyright analysis, package analysis, meta data extraction, and MIME type identification.

Presented will be an overview of how the FOSSology framework works, how Debian can benefit from it and participate in the project, and a short

demonstration of FOSSology analysis of Debian. This talk may especially interest Debian package maintainers and members of the ftpmaster, legal, and QA teams.

1.1.21 Free software for replacing Skype

David Alexander Sugar ⟨dyfet@gnutelephony.org⟩

The GNU Telephony secure calling project, built around GNU SIP Witch and the GNU ZRTP stack, offers the means for anyone to create and deploy network scalable and secure peer-to-peer VoIP collaboration solutions to enable privacy without the need for a central service provider, undocumented protocols, or proprietary software to achieve these goals. Our overall vision is to facilitate both solutions that are privately built, such as for organizations that wish to have secure communication as a foundation, and especially which can be autonomously assembled bottom-up directly by users over the public Internet as a full public alternative to Skype, using only free software and that depends purely on existing DNS for user lookup, rather than requiring a mediating service provider.

1.1.22 GPLv3: Better Copyleft for Developers and Users

Bradley M. Kuhn ⟨bkuhn@ebb.org⟩

Many developers have not yet investigated the new suite of GPLv3 licenses. This presentation helps developers already familiar with GPLv2 and LGPLv2 to improve their understanding of GPLv3, LGPLv3, and AGPLv3. This presentation covers changes in the patent provisions, improved license compatibility, improved source code provision terms, the internationalization of GPL and many other changes.

1.1.23 Group photo

Aigars Mahinovs ⟨aigarius@debian.org⟩

1.1.24 Handling Debian bugs with SD

Christine Spang ⟨christine@debian.org⟩

Simple Defects (SD) is a command-line distributed bugtracker. If you've heard of Bugs Everywhere or Ditz, you have the general idea.

Bugs Everywhere and Ditz might be nice for tracking bugs for your own new project, but they're not very useful for working with Debian. What may surprise you is that SD is!

SD has a special feature that other distributed bug trackers lack: the ability to synchronize bugs back and forth with bug trackers that are external to itself. It does this through pluggable adaptors that work with SD's "foreign replica" interface. This allows you to grab bugs matching some criteria from a foreign source, work on them locally (even offline) with all the power of SD, and then send the updates back to the centralized bugtracker.

This talk will present my work on an SD adaptor to interface with our beloved Debbugs, and how it can make you, Debian contributors, more productive when dealing with bugs.

1.1.25   How Government can Foster Freedom in Technology

Hon. Gale Brewer ⟨dkg-galebrewer@fifthhorseman.net⟩

Hon. Gale A. Brewer, former Chair of the Committee on Technology in Government, current Chair on Committee on Governmental Operations will discuss the ways that government can foster freedom in technology.

1.1.26   How We Can Be the Silver Lining of the Cloud

Eben Moglen ⟨None⟩

Freedom and privacy are endangered in the cloud, but we can save them. In this, the second part of my analysis of Freedom in the Cloud[0], I show how the Debian community can play the central role in changing the future of Net, again. Contact: press@softwarefreedom.org Software Freedom Law Center[1] (Abbreviated) Bio and link to a full version on the SFLC website[2]

1.1.27   Integrating new people in packaging teams

Ana Beatriz Guerrero López ⟨ana@debian.org⟩

I will share my experiences within the Qt/KDE team and hopefully this will lead to a discussion with other members of others teams about what worked with them and what did not work while trying to attract new members.

1.1.28 Interesting network features of your Debian system

Guido Trotter ⟨ultrotter@debian.org⟩

Networking is still very much a domain in which big proprietary computing equipments dominate. The Linux kernel, though, has some very neat networking features, and is increasingly being used to replace those systems with something more open and more adaptable to "strange" needs. During this talk we will explore some less-known Linux networking features, such as NBMA networks over unbound GRE tunnels, policy routing, or anycast that will allow systems administrators to take control of the network, and even sometimes make something slightly crazy with it, and make some of those networking people, who only trust those closed machines on which they're certified, a bit (or a lot) nervous.

1.1.29 Job/Gig Recruiting Session

Jimmy Kaplowitz ⟨jimmy@debian.org⟩

Many highly qualified people who come to DebConf are between jobs, looking to switch jobs or home countries, or are freelancers seeking additional contracts. This is true every year, but even more so in 2010 due to the global recession. Many others work at companies that could use the professional services of DebConf attendees. This event is intended to bring the two groups together so that job or contract opportunities can be discussed face-to-face. Another potential topic of discussion is how to search for work in various unfamiliar cities or countries.

1.1.30 Keysigning Information/Discussion Session

Aníbal Monsalve Salazar ⟨anibal@debian.org⟩

We will discuss how keysigning will work at DebConf10. You may also want to join us for the Classical Keysigning Party on the evening of August 6th.

1.1.31 Lightning Talks

Micah Anderson ⟨micah@debian.org⟩

Lightning Talks are 11 five-minute talks in a 60-minute time-slot!

1.1.32   Lintian BoF

Russ Allbery ⟨rra@debian.org⟩

Discussion of the current status of Lintian, the current Lintian architecture, desired future architectural and testing changes, and how you can help.

1.1.33   Making Debian Rule, again.

Margarita Manterola ⟨marga@debian.org⟩

An exploration into what we are not doing well enough to make Debian the best of the best, and how we could go about making things better.

1.1.34   Monkeysphere

Daniel Kahn Gillmor ⟨dkg@fifthhorseman.net⟩, Greg Lyle
⟨greg@stealthisemail.com⟩

Monkeysphere is a project to extend the OpenPGP Web of Trust into as many domains as possible, effectively supplanting hierarchical certification infrastructure like X.509, and restoring control over authentication and identification to the communications peers themselves and their own legitimately trusted introducers.

We will discuss how Debian can take advantage of Monkeysphere and related ideas, and how packagers can make sure their packages can make use of it.

1.1.35   Multiple repository bootstrapping

Neil Williams ⟨codehelp@debian.org⟩

Multistrap is a multiple repository bootstrapping package which uses apt to resolve the dependencies and a series of configuration files and custom scripts to prepare a complex chroot or a bootable rootfilesystem in a single process. This talk describes the current multistrap process.

1.1.36   News on Debian Autobuilding

Philipp Kern ⟨pkern@debian.org⟩, Kurt Roeckx ⟨kurt@roeckx.be⟩

1.1.37 Packaging Python modules and applications

Piotr Ożarowski ⟨piotr@debian.org⟩

This talk briefly introduces current and future Python packaging helper tools. Python 3.2 allows sharing source code files among multiple different versions of Python interpreter, new dh_python will take advantage of it.

1.1.38 Patent Absurdity

John Sullivan ⟨john@wjsullivan.net⟩

"Patent Absurdity: How software patents broke the system" explores the case of software patents, the history of judicial activism that led to their rise, and the harm being done to software developers and the wider economy. The film, directed by independent filmmaker Luca Lucarini and funded by a grant from the Free Software Foundation, is based on a series of interviews conducted during the Supreme Court's review of in re Bilski, a case that could have profound implications for the patenting of software. The film was produced and edited using free software tools on GNU/Linux, and is being distributed in the free Ogg Theora format. I had a part in its creation as part of my work at the FSF, and could be available to answer questions and give some background both about the process behind the film and an update on the work of the FSF's End Software Patents campaign. The Bilski decision should have come down by the time Debconf happens, so there should be much to talk about with regard to what the outcome means for patents, free software, and Debian. The film is about 30 minutes long.

1.1.39 Querying our BTS with python-debianbts

Bastian Venthur ⟨venthur@debian.org⟩

1.1.40 R Users and Maintainers Bof

Don Armstrong ⟨don@debian.org⟩

R is the GNU clone of S, and is heavily used in statistics and multiple scientific disciplines. Discussion about maintenance, packaging, and usage coupled with meeting others who use R.

1.1.41   RC bug fixing + NMU = Fun

Stefano Zacchiroli ⟨zack@debian.org⟩

In recent release cycles, it has become more and more evident that a "selfish" approach in which maintainers only care about their own (team maintained) packages does not scale. To keep up with our principle of releasing RC-bug-free software we all need to care about the packages of other, in particular when they are RC-buggy. That, of course, involves actions such as NMU-ing properly, but timely too.

This BoF aims will bring together people that care about keeping the RC bug count low not only for what concerns their own packages, but also when it comes to packages of other fellow DDs. Topics of discussion include:

- How to spot packages that need NMU from other maintainers. - Official guidelines for when and how do NMUs (devref §5.11.1). - Best practice and invidual work-flow for NMU campaign (the guidelines used by the BoF proponent are in the links section). - How (and if?) communicate about NMU campaigns to get other fellow developers involved. - How to train new/forthcoming Debian developers to care more about packages other than theirs. - etc.

1.1.42   SPI BOF

Michael Schultheiss ⟨schultmc@debconf.org⟩

Debian is an Associated Project of SPI and this BoF is a chance to meet the SPI Board members who are present at DebConf 10 and learn more about SPI and why you should become or continue to be involved in SPI.

1.1.43   So you want to host DebConf12?

Jimmy Kaplowitz ⟨jimmy@debian.org⟩, Gunnar Wolf ⟨gwolf@debian.org⟩

1.1.44   Stable Release Management

Philipp Kern ⟨pkern@debian.org⟩

1.1.45   Stargazing

Cameron Hummels ⟨greenphreak@gmail.com⟩

At 9:00PM the Columbia University Astronomy Department will host a guided stargazing event, which includes use of several different optical telescopes on

their roof. There will be astronomers on hand to answer any questions about the stars, telescopes, or any other topics of scientific interest.

For more information on the Columbia Astronomy Outreach program, see http://outreach.astro.columbia.edu/

1.1.46 State of Debian (based) Linux live systems in 2010

Michael Prokop ⟨mika@debian.org⟩

This talk will provide an overview of the current state of Linux live systems. How do the various ways to boot a live system work and which ways exist? Which ways exist to build a live system and what steps are involved? Where are chances for improvement and what's needed for the future of live systems? What is Grml and its unit testing framework and how could it be used for Q/A stuff within Debian?

1.1.47 Statistical Machine Learning Analysis of Debian Mailing Lists

Hanna Wallach ⟨hmw26@srcf.ucam.org⟩

In this talk, I will discuss the use of state-of-the-art machine learning techniques to analyze Debian mailing lists in order to discover political, social, and technical patterns that could be used to inform project decisions. I will concentrate on a class of techniques known as statistical topic models, which automatically infer groups of semantically-related words, known as topics, from word co-occurrence patterns in documents. The resultant topics can then be used to detect emergent areas of technical activity, identify subcommunities, and track trends over time. In addition to providing a brief overview of statistical topic models and their application to Debian mailing list data, I will present examples of topics inferred from Debian mailing lists, as well as some preliminary political, social, and technical findings discovered via these topics.

1.1.48 Supporting Debian machines for friends and family

Francois Marier ⟨francois@debian.org⟩

How to provide friends and family with reliable Debian-based machines without spending all of your evenings maintaining them.

1.1.49   Swimming upstream

Lars Wirzenius ⟨liw@liw.fi⟩

Debian is a huge project, with tens of thousands software packages, a thousand people, and hundreds of communication forums. Koha is large, old, somewhat crufty, free software system for running libraries (the kinds with books and stuff).

Koha upstream wants it packaged for Debian, and eventually included in Debian. Because of the sizes of both projects, and the established cultures and policies and habits, things are not always straightforward.

What are the challenges Koha faces? What can Koha do to make integration into Debian better? What could Debian do? What lessons can be learned to help other upstreams?

1.1.50   The Challenges of Licensing in Debian

Daniel German ⟨dmg@uvic.ca⟩

In this presentation I'll describe some of the major issues affecting both free software developers and Debian package maintainers, and some results that we believe can improve the license compliance of both packages and distributions, such as Debian.

1.1.51   The Future of the Debian Volatile Archive

Philipp Kern ⟨pkern@debian.org⟩

1.1.52   The Nexenta Project: OpenSolaris + apt + GNU

Anil Gulecha ⟨anil@nexenta.org⟩

The Nexenta project started in 2005 as a port of Ubuntu LTS (Dapper Drake 6.06) to OpenSolaris. It is now the most popular non-Sun distribution based on OpenSolaris. It also forms the basis of multiple community and commercial distributions. This talk traces the history of the Nexenta project, it's relationship with the Debian project, and the opportunities for collaboration.

1.1.53   Travel Sponsorship BoF

Pablo Ariel Duboue ⟨pablo.duboue@gmail.com⟩

Year in, year out we use a similar process to chose which attendees to give money to help offset travel expenses. There is always discussion about improving the process but when the time comes it is very difficult to do it plus improve. Let's start a year earlier this time! Anybody that helped rank people this year or previous year is expected to attend this meeting. People who requested travel sponsorship are also welcomed as their feedback about the process will also enrich the discussion.

1.1.54 Upstart in Debian

Scott James Remnant ⟨scott@ubuntu.com⟩

A discussion about what needs to be done for Debian to adopt the Upstart replacement init system

1.1.55 What git-buildpackage (hopefully) can do for you.

Guido Günther ⟨agx@debian.org⟩

A distributed version control system like Git can ease common packaging tasks like importing new upstream versions or backporting. The tutorial will give an introduction into maintaining Debian packages in Git supported by the tools from git-buildpackage.

1.1.56 What if.... we could automatically test our textual instructions?

Pablo Ariel Duboue ⟨pablo.duboue@gmail.com⟩

A common problem in any complex computational environments are stale instructions. Software systems change but their related instructions are left unmodified. This problem is made further complicated in Debian by the large number of human languages supported by the project.

Branavan et al. (2009) describe an automated way to translate human, textual instructions into executable actions. [1]

Their approach involves reinforcement learning with no human intervention, by probing alternative interpretations of the instructional text in a virtual machine. The computer learns which different actions (such as "choose third option from the menu") correspond to different English words in the instructions. Their system also learns how to re-order instructions (as text presentation doesn't necessarily imply execution order, e.g. "click done after

selecting /dev/usb"). They are also making their code available (targeted towards a MS Windows VM). [2]

In this talk, I will sketch their approach, show a prototype of a Debian-adapted system and seek out collaborators to put this technology to use in different parts of the project, especially in d-i and i18n.

[1] Reinforcement Learning for Mapping Instructions to Actions ACL 2009 [Best paper award] S.R.K. Branavan, Harr Chen, Luke Zettlemoyer, Regina Barzilay. ⟨a href="http://people.csail.mit.edu/branavan/papers/acl2009.pdf"⟩http://people.csail.

[2] ⟨a href="http://groups.csail.mit.edu/rbg/code/rl/"⟩http://groups.csail.mit.edu/rbg/code/r

1.1.57   Why is GNU/Linux Like a Player Piano?

Jon Anderson Hall, Esq. ⟨maddog@li.org⟩

One of the first programmable devices was the Jacquard loom, which used "punched cards" to control fabric designs. One of the next main programmable devices was the player piano (pianola) that used a spool of punched paper to control a mechanical device. There are a lot of similarities between Free Software and the Player Piano, and a lot of lessons to be learned about copyrights and patents, including the person who first invented spread-spectrum radio. This talk illustrates these similarities and lessons, leading up to a discussion of the Opus I organ at Trinity Church on Wall Street, perhaps the greatest organ in the world, driven by GNU/Linux.

1.1.58   X in squeeze

Julien Cristau ⟨jcristau@debian.org⟩, Cyril Brulebois ⟨kibi@debian.org⟩

The graphics stack has once again seen massive changes in squeeze. After going for auto-configuration in lenny, squeeze introduces kernel mode setting. The input subsystem was not left alone, with support for device hotplug and multi-pointer X. This talk will cover what's new in X in squeeze, what that means for the Debian packaging, and what we can expect in the future.

1.1.59   notmuch

David Bremner ⟨david@tethera.net⟩

Notmuch is a mail indexer and user agent by Carl Worth that seems to have caught the fancy of several "Debian people". We will talk about and/or hack

on prioritizing existing patches for the next release, identifying missing functionality and nailing down the design of some of the more complex proposed features like better maildir integration, synchronization (including ideas like tag logs). It could also be useful to brainstorm about our patch management process and in general whether technical or social improvements could be made to the development process.

1.1.60   use Perl;

Gregor Herrmann ⟨gregoa@debian.org⟩

The pkg-perl team will again take the opportunity to meet in person for discussing current topics and planning future work.

Items for discussion and work are collected at http://wiki.debian.org/Teams/DebianPerlGroup/OpenTasks

1.2   Ad Hoc

1.2.1   ARM porters friendly talk

Hector Oron ⟨zumbi@debian.org⟩, Steve McIntyre ⟨93sam@debian.org⟩

Debian ARM porters meeting for Open discussion on improvements Debian porters could do to Debian/ARM architecture.

This talk was merged with Steve's BoF request for working on the arm/armel architecture.

1.2.2   CUT BoF

Joey Hess ⟨joeyh@debian.org⟩

This BoF will be used to explore the feasibility of the Constantly Usable Testing concept. Encouraged to attend: Representatives from Release Team, FTP Team, Security Team, d-i and CD teams, and Publicity Team. And anyone with an interest in forming a CUT Team.

1.2.3   DebConf & Debian

Holger Levsen ⟨holger@layer-acht.org⟩, Stefano Zacchiroli ⟨zack@debian.org⟩

Discuss why (and how far) DebConf and Debian are two seperate entities. Discuss future.

1.2.4   Debian Live Team Meeting

Daniel Baumann ⟨daniel@debian.org⟩

1.2.5   Debian Merchandise BoF

Luca Capello ⟨luca@pca.it⟩

Discussion about which kind of merchandise Debian-associated people/community produce, also to let people attending conferences/meetings/whatever how to get merchandise. Currently Alexander Formorer, Debian Brasil, debian.ch and Debian UK produce merchandise for Debian.

1.2.6   Distributed Bug Tracking BoF

Don Armstrong ⟨don@debian.org⟩

1.2.7   Freeing the Cloud, one (small) service at a time

Francois Marier ⟨francois@debian.org⟩

Introducing a new project to provide an open and federated alternative to the centralized Gravatar avatar hosting service.

1.2.8   The Unity Desktop Environment

Mark Shuttleworth ⟨marks@debian.org⟩

Mark will present the work that Canonical has done on netbook user interfaces. The netbook is a significant force in Linux adoption, and Canonical's design team has made netbook user experience its priority. This talk will cover work done to improve the experience people have with Linux on small screen laptops, and examine the possibilities for convergence between netbook and large-screen desktop environments.

1.3   Community Outreach

1.3.1   Beyond Sharing: Open Source Design

Mushon Zer-Aviv ⟨mushon@shual.com⟩

Keywords: design, collaboration, interface, community

"Design by committee", "too many cooks in the kitchen" and other terms have been used derogatively to imply that the creative process breaks when it involves too many individuals. At the same time, the software world has been completely revolutionized by the open source, networked collaborative process. When it comes to graphic and interaction designs (two fields critical to software development) Open Source software seem to lag behind the more conventional production models. It seems like in design the Open Source community have not managed to nourish robust collaboration models. One could argue that even successful free software design projects like Firefox or Ubuntu are often developed in a more traditional process. What are the challenges of networked collaboration in the creative process? Can and how might they be solved? Or is it just that designers don't like to work together?

I will be tackling these questions in the light of my collaborative work as founding designer of ShiftSpace.org and my educational research leading the Open Source Design class at Parsons' AAS Program in Graphic Design.

6mins version of this talk is available here: http://www.mushon.com/2009/12/18/shortaudio-my-open-source-design-slides/

*Disclaimer: I am not a Debian community insider, that's a part of what I want to talk about.

1.3.2   Collaboration between Ubuntu and Debian

Jorge O. Castro ⟨jorge@ubuntu.com⟩

I would like to cover how maintainers from both projects can coordinate over patches, bug, version control systems, roadmaps, etc. I am also going to include some lessons learned about what it's like to be a derivative distro of Debian, so I will use examples from the past where things went well (or didn't go well at all!) so that both Debian Developers and derivative's from Debian can learn what works and what doesn't.

1.3.3   Debian at a large Dutch University

Frank B. Brokken ⟨f.b.brokken@rug.nl⟩

The paper and and presentation I propose will concentrate on the importance of Debian products for the University of Groningen, one of the major Dutch Universities.

To summarize:

- The university's Center of Information Technology (CIT) is responsible for the correct functioning of the University's Information Technology facilities. This involves making available 'standard working environments' for its 5000 staff members and 20,000 students, monitoring and maintaining the integrity of its 10Gb Internet connection as well as the computers that are responsible for providing the facilities that are considered essential for the university.

The University uses various means to protect and monitor the well-being of its computers. Among other means the Stealth File Integrity Control program, a Debian package, is used to monitor the file integrity of its core computer systems.

- The standard working environment offered by the university to its employees and students has always been a non-Linux system. Fortunately that has changed by now and Linux systems are now deployed in more and more departments of the university. This 'Linux Working Place' (LWP) as it is affectionately called is strictly speaking not Debian but Ubuntu, but the close links Ubuntu has with Debian result in a de-facto use of Debian (see also below). In addition to this 'official' acceptance of Linux, several departments (Astronomy, Computer Science, Artificial Intelligence, to name a few) have been using Debian Linux for a long time.

- Various programming languages are actively being taught by the University. Since the mid-80s I have organized a course on the C programming language, augmented with C++ since the early 90s. The addition of C++ resulted in the 'C++ Annotations', which became the main text-book for that course. The C++ Annotations have also found their way into Debian and are used by many individuals and organizations on a world-wide basis for teaching and honing knowledge about C++.

- Students reaching the advanced levels of the C++ course use a parser generator specifically designed to be used with C++ when when designing and implementing their own grammars. This parser generator is Bisonc++, also a Debian package.

- Students of my class 'Information Security' use classes and templates of my 'Bobcat' software library to implement algorithms requiring computations with integral values of unlimited sizes. They usually prefer using these specially designed classes over the raw use of the underlying OpenSSL library functions. Bobcat is not only used in this context, but many C++ programs developed during the C++ course or used by the C++ course students rely on Bobcat. Bobcat is also used in other sections of the University. The departments of Computer Science and Artificial Intelligence customarily have Bobcat installed on their systems as well.

- Various generic tools (Icmake for program development, Xd for fast directory changes and other Debian-based tools) are used by students and staff-members to simplify their daily activities. Spin-offs of these projects were developed or are being developed right now by others, like 'ccbuild' (Bram Neijt) and 'flexc++' (Jean-Paul van Oosten, Richard Berendsen, and myself). Moreover, the Debian Developer George Danchev visited us last fall when the CIT organized a course on Debian Package Construction.

1.3.4 FLOSS Manuals: A Vibrant Community for Documentation Development

Andy Oram ⟨andyo@oreilly.com⟩

Started in 2005, FLOSS Manuals has been the locus for over 40 volunteer-produced books and ties together a worldwide community of over 1,000 people who care about free software and free documentation. This presentation describes the value of FLOSS Manuals, its development processes and leadership style, and its strengths and weaknesses as seen from its key organizers.

1.3.5 FSF's campaigns for freedom

John Sullivan ⟨john@wjsullivan.net⟩

The FSF works for user freedom on several fronts, including activist political campaigns, educational resources, and support for software development. All of these areas intersect with Debian – activist campaigns like Defective by Design, PlayOgg, and Windows 7 Sins are supported by many Debian participants and relate to the goals and values of the DFSG. Resources like the Free Software Directory, Hardware Database, and Licensing & Compliance Lab provide important information for free software users and developers. Through the GNU Project and the High Priority Projects list, the FSF directly supports upstream free software development. As the Operations Manager at the FSF, I can give an overview of these projects, and focus on the most timely ones. It's labelled as a lecture, but I would hope to have a significant amount of discussion, including thoughts about our existing efforts and suggestions for what more we could do.

1.3.6 Google Summer of Code 2010 at Debian

Obey Arthur Liu ⟨arthur@milliways.fr⟩

This event will gather the admins, mentors and students of the 2010 Google Summer of Code at Debian to present our organization work, the students, their projects and how they are doing.

1.3.7   Pedagogical Freedom

Jonah Bossewitch ⟨jonah@bossewitch.net⟩

Educational contexts are powerful testimonies to the value and importance of free software. This panel will discuss various ways that Debian and free software are currently being used to help improve education, in the broadest sense of of the term. Free software is being used effectively across educational institutions, libraries, and grassroots advocacy efforts. We will examine various ways that new media and technology are being used to transform teaching, learning, access, and communication.

   * Jonah Bossewitch (http://alchemicalmusings.org), Technical Architect at CCNMTL (http://ccnmtl.columbia.edu) and doctoral student in Communications at Columbia's School of Journalism. * Mallory Knodel (http://malloryknodel.net), May First/People Link (http://mayfirst.org) * Professor Matt Curinga (http://matt.curinga.com), Adelphi University * Professor Alycia Sellie (http://www.brooklyn.cuny.edu/pub/Faculty_Details5. Brooklyn College * Ahamed Shameem (http://www.projectfedena.org), Fedena is an Open Source School Management System based on RoR technology. It is a one stop solution for all teaching and school management related issues and it helps to manage all school related activities online.

1.3.8   Project Caua: Private Sector, Environmentally Friendly Jobs with Free Software

Jon Anderson Hall, Esq. ⟨maddog@li.org⟩

Project Caua is an Open project to create millions of private sector, environmentally friendly jobs in urban areas of Latin America utilizing FOSS, with implications of creating hundreds of thousands of jobs in North America and millions more around the world.

   In addition, Project Caua will open an avenue for free (as in beer) wireless Internet to help defeat the digital divide, and to provide low-cost training to get even more people off unemployment and create taxpayers.

   The specifications for Project Caua can be found at www.projectcaua.org.

   While Project Caua is distribution neutral, Debian would be a natural choice as the underlying distribution and sample implementation. This talk

will outline Project Caua and hopefully open the discussion for actual implementation details.

1.3.9 Revealing Errors

Benjamin Mako Hill ⟨mako@debian.org⟩

This talk provides a humorous description of an argument in favor of Debian, and free software more generally, based on what I call "antifeatures:" functionality that users hate and that technology developers will often charge users to not include. From DRM to crippled non-free operating systems to digital cameras, I will show off many of the worst antifeatures and, describe how Debian and free software both makes them impossible and helps users work around them.

1.3.10 Steps to adopting F/OSS in government

Andy Oram ⟨andyo@oreilly.com⟩

This talk looks at both successful and unsuccessful attempts to adopt open source software (the term is preferred to "free software" in government) and lays out some principles for people inside and outside government agencies to promotes its use. In particular, I will discuss the assessment of costs and benefits done for the city of Munich in Germany, the creation of Forge.mil by the Department of Defense, and the process used to choose OpenOffice.org for the state of Massachusetts (a project that was defunded). The importance of using explicitly political arguments for the adoption of open source software, and for agency managers to understand the community and workings of open source at a deep level, will be stressed.

1.4 DebCamp

1.4.1 Videoteam Training

Eric Dantan Rzewnicki ⟨eric@zhevny.com⟩

1.4.2 Videoteam Training

Eric Dantan Rzewnicki ⟨eric@zhevny.com⟩

1.5   Enterprise

1.5.1   Enterprise Infrastructure BOF

Sam Hartman ⟨hartmans@debian.org⟩

A BOF to explore enterprise infrastructure software such as LDAP, Kerberos, Samba etc and how these packages can work better together to make Debian be a better building block for groups of computers

1.5.2   Managing Debian Installations

Michael Banck ⟨mbanck@debian.org⟩

Fully Automatic Installation (FAI) is a well-known project to automatically install multiple machines using Debian. Those machines can further be kept uptodate via the "softupdate" mechanism. This has been proven very successful for big cluster installation at acedemic institutions where the adminstrators have a good Linux/Unix background. However, for small/medium business or smaller acedemic work groups, FAI might be to obscure or difficult to setup. The GOsa project provides an LDAP-based web administration tool to install, configure and administrate machines via FAI. GOsa makes furthermore possible to administrate users and groups via the same LDAP backend, and to some extent user sessions. This makes it possible for people who are no specialists to administrate smaller sets of users and machines via a centralized web frontend.

This talk will present FAI and GOsa and explain how they work together and what is needed to install and adminstrate Debian machines via GOsa/FAI.

1.5.3   Moonshot: Federated Authentication beyond the Web

Sam Hartman ⟨hartmans@debian.org⟩

The world is filled with consumer and web-focused federated authentication solutions. However as federations grow to hundreds of identity providers, enterprises begin to see strain of the current design. Moonshot combines four key security technologies (EAP, SAML, RADIUS and GSS-API) to address these scaling challenges and to bring federation to any software, not just the web browser. This talk describes the Moonshot technology as well as the ongoing community-driven free-software effort to realize the technology. One key focus is the role of Debian in particular and distributions in general in driving coherent technology evolution.

1.5.4   Puppet BOF

Matt Taggart ⟨taggart@debian.org⟩

From the puppet upstream site, "Puppet is an open source data center automation and configuration management framework. Puppet provides system administrators with a simplified platform that allows for consistent, transparent, and flexible systems management." We will discuss using puppet on Debian, the state of puppet and related packages in Debian, and share best practices and puppet modules.

1.5.5   Using Debian for Enterprise Infrastructure

Russ Allbery ⟨rra@debian.org⟩

A discussion of Debian as the server platform of choice for enterprise-wide infrastructure, using Stanford University as a case study. Topics covered will include packaging and package modification for site-specific needs, challenges in packaging software for local deployment, repository management and local documentation issues, and ways that Debian can better support this use case.

1.5.6   Using Debian to provide services to thousands

Faidon Liambotis ⟨paravoid@debian.org⟩

The talk will describe the services that the Greek Research and Technology Network (GRNET) provides to its customers in the Greek academic community, with a target group of hundred of thousands of users. It will focus on the challenges faced when providing these services and the almost exclusive use of Debian to solve those.

GRNET offers a multitide of services to its customers, both on the Internet and computing fronts, on a multi-vendor environment. Over two hundred physical servers and twice as that on virtual servers are being operated by a handful of people, all but few running Debian.

Topics that will be mentioned are ranging from networking monitoring and identity to virtualization and cloud. The use and integration of multiple free software components on a Debian platform to accomplish those will be described along with a focus on high availability and automation of system administration tasks out of necessity and the ways and tools (e.g. d-i, puppet, Ganeti) to accomplish that.

Faidon Liambotis is on the Server & Services team of the Greek Research and Technology Network's Network Operations Center. He is also a Debian Developer.

1.6   Evening and Social

1.6.1   "Sita Sings the Blues" Screening and Q&A

Karl Fogel ⟨kfogel@red-bean.com⟩

"Sita Sings the Blues" is a popular film released under a Creative Commons Attribution-ShareAlike license. The filmmaker, Nina Paley, consciously chose a Free license in order to achieve the same dynamics for her film as software has in the free software world, and she succeeded.

By the end of 2009, "Sita Sings the Blues" was the best-reviewed new release in the country (http://moviereviewintelligence.com/index.aspx?BID=27&RID=805&CID=0) and audiences love it. We will hold a talk afterwards with the filmmaker and myself (Nina Paley is now artist-in-residence at our non-profit QuestionCopyright.org, and we helped her do the free release), both answering questions from the audience and talking about how free software methods can work for artists in general.

1.6.2   Cheese and Wine party

Christian Perrier ⟨bubulle@debian.org⟩

This is the now traditional Cheese and Wine party of Debconf. As usual, things are simple: people do their best to bring cheese, wine and other beverages from all around the world, and share them.

1.6.3   DC10 runs in Central Park

Christian Perrier ⟨bubulle@debian.org⟩

Running in Central Park is somewhat mythical for people who like running/jogging/etc. So, the crazy Debconf runners couldn't miss that opportunity. We plan to gather everybody interested in this, have them "sport suit up", then go running around the Reservoir. We really intend to be a fun moment, not a competition, nor a hard practice for marathoners. Just a good moment to share with people who like to move their body.

1.6.4 Games!

Matthew Johnson ⟨mjj29@debian.org⟩

1.6.5 RCBC - Release Critical Bugs of the Conference

Stefano Zacchiroli ⟨zack@debian.org⟩

Debian Squeeze will be released well before DebConf10, that's for sure, probably next week *g*.

In the (unlikely) case that won't happen, having a conference-long BSP (Bug Squashing Party) to help fixing the (few) remaining RC bugs would be useful.

So, join the RCBC contest, the contest in which gain fame and glory being the one who fixed the most RC bugs during DebConf10!

Technical details are still to be determined, but there will be:

(1) Technical ways to submit a specific RC bug fix (downgrade, NMU, close, etc.) for the contest. (2) Prizes offered by DebConf organization and/or brought collectively by other DebConf participants. (3) Clear rules to break ties, decide how many winner we'll have, etc. (4) An actual public event, towards the end of the conference, where the winner will be acclaimed by the masses. (5) Honor, fame, glory for the winners.

In the [likely] case Squeeze will be ready before DebConf10, no problem, we'll do the same for the next Debian release!

[And blameŴthanks to micah for tricking me into propose/organize this!]

1.7 Java

1.7.1 A battle cry for a system-level JVM in Debian

Pablo Ariel Duboue ⟨pablo.duboue@gmail.com⟩

If you are running an ant build, trying to get it right, how many times do the Java API methods Just-in-Time compile from Java bytecodes into machine code? For every invocation of /usr/bin/ant, a JVM is spanned and relevant methods are compiled. This is a clear waste of work. Moreover, just-in-time compilation is one of the main reasons of Java perceived sluggishness. As Java is enjoying a sort of renaissance in the Free Software world after the release of OpenJDK, we can expect more and more times our system will be just-in-time compiling yet another copy of java.util.regexp.Pattern.

A better solution would be to have system-wide JVM where well behaved (also known as "Debian packaged") programs and libraries can co-exist.

Luckily, researchers at Sun Research Labs were working five years ago in project Barcelona [1] (now concluded) which produced a research prototype of a system level JVM (they call this a Multi-tasking Virtual Machine or MVM for lack of a more confusing name) to address this very issue.

Their approach resulted in JSR 121 (isolates) [2] for JVM extensions allowing multiple Java applications (classes with a public static void main) to be executed concurrently in the same JVM.

Implementing JSR 121 inside OpenJDK is clearly outside the scope of Debian but even in the presence of a JVM with isolation capabilities, there is plenty of system-specific work that needs to be tackled (what happens with /usr/bin/java? it is really system-wide or user-specific? how "well behaved" do the applications need to be?).

In this talk, we will summarize relevant bits of project Barcelona's ample list of publications and focus on simple, JSR 121-alike simulations to show what can we expect from having this technology incorporated into Debian.

The hope is that we can spearhead the use of isolates within GNU/Linux systems so as to contribute back this knowledge to other distros.

[1] http://research.sun.com/projects/barcelona/ [2] http://jcp.org/aboutJava/communityproce

1.7.2   Debian-Java BoF Session

Pablo Ariel Duboue ⟨pablo.duboue@gmail.com⟩

The Debian Java Maintainers [1] team has 50+ members and currently maintains close to 500 packages [2], including OpenJDK [3] and the Java Policy [4]. In this birds-of-a-feather session we will discuss issues related to Java and Debian. Anybody interested in learning more about it is welcome to attend. People who maintain Java packages outside the team maintenance umbrella are highly encouraged to attend as their feedback is most appreciated.

[1] https://alioth.debian.org/projects/pkg-java/ [2] http://pkg-java.alioth.debian.org/cgi-bin/pet.cgi [3] http://packages.debian.org/source/sid/openjdk-6 [4] http://www.debian.org/doc/pa manuals/java-policy/

1.7.3   Debian-java policy changes

Matthew Johnson ⟨mjj29@debian.org⟩

Until this year Debian Java policy has been stagnant for quite a long time. With the changes in the Java ecosystem, particularly the arrival of openjdk,

a lot of changes are necessary. Niels Thykier has pushed through a lot of the changes which had been discussed, but lost momentum.

This talk covers those changes which have been made and lead discussion into what policy changes are further desirable. Changes that Niels made were largely to bring policy in line with current practices. We will also review what things are not in policy but which are part of current packaging practices.

Discussion on future policy changes will . We will also review what things are not in policy but which are part of current packaging practices.

Future policy changes to be discussed will focus on simplifying library transitions, dependencies and other issues.

1.7.4   Packaging with javahelper

Matthew Johnson ⟨mjj29@debian.org⟩

Programs and libraries written in Java have a lot of special packaging requirements in common. When I started packaging Java programs I was annoyed at the lack of tools to automate some of these tasks and hence Javahelper was born.

Javahelper provides tools to help with the workflow right from processing upstream releases (which can be tricky with some Java upstreams) through building packages where upstream doesn't have a sane build system to generating packaging metadata and installing files into packages.

This talk gives an overview of the dh7 and cdbs integration, how to use the various helper tools available in javahelper and finishes with example packages of a selection of simple Java programs and libraries.

The aim of Javahelper is to provide tooling to implement Debian Java policy, automate common tasks required by Java packages that aren't provided for by debhelper and to simplify the packaging of Java software.

1.7.5   The Free Java Jigsaw Puzzle

Tom Marble ⟨tmarble@info9.net⟩

Extending the modularity of Free Java to match Debian.

1.7.6   The Java Packaging Nightmare

Torsten Werner ⟨twerner@debian.org⟩

The talk shows the problems of packaging Java code in a way that the result conforms to minimal packaging standards. Sometimes code does not get build

completely from source, sometimes non-free software gets used and sometimes the license terms are not correct. Ways to improve that situation will be shown on the example of Maven.

1.8 Media and Art

1.8.1 Brave new Multimedia World

Reinhard Tartler ⟨siretart@debian.org⟩, Eric Dantan Rzewnicki ⟨eric@zhevny.com⟩, Adrian Knoth ⟨adi@drcomp.erfurt.thur.de⟩

1.8.2 Conference Video

Eric Dantan Rzewnicki ⟨eric@zhevny.com⟩

Video recordings and live internet streams of a growing number of Gnu/Linux and Free Software conferences around the world are being produced by teams of volunteers using largely donated resources and Free Software running on Linux. This talk discusses the hardware and software used for a variety of the productions in which DebConf videoteam members have participated. The software covered includes dvswitch and tools from xiph.org. Challenges and experiences with regard to quality sound, lighting, volunteer organization and workflow management will also be touched upon.

1.8.3 Debian as a home TV etc. management system

Kristoffer H. Rose ⟨krisrose@debian.org⟩

I hope that we can have a discussion of how installation of MythTV (and related systems such as freevo) can be even better on Debian. Possibly it would be nice to investigate whether shared infrastructure is appropriate such as permitting "following" other people's recording selections, permitting social interactions in the middle of viewing, etc.

Also we are in dire need of some easy way to integrate music collections properly with the multiple music managers...

1.8.4 Hands-On DebConf Video Training

Eric Dantan Rzewnicki ⟨eric@zhevny.com⟩

The tools we have are pretty cool. Describing them and their use in a one-way presentation is good and useful. But, the complexity of the system is better understood through getting one's hands on it

1.8.5   Improving Debian's Multimedia Experience

Reinhard Tartler ⟨siretart@debian.org⟩

Currently, many multimedia related packages are still unpackaged, or packaged but not in debian. Our users therefore have to choose from a number of 3rd party distributors or compile software from source which leads to a poor user experience.

This BoF discusses the needs, requirements and interest of 3rd-party archives that contain multimedia related packages. The goal is to identify the current biggest multimedia related problems in debian and tries to form suggestions on how to improve the situation.

1.9   Science

1.9.1   Debian Science Round Table

Michael Banck ⟨mbanck@debian.org⟩

Scientific communities are a big part of Debian's ecosystem and one of the original driving forces of Debian and Free Software in general. This round table will bring together the leaders of the respective communities (physics, biology, chemistry, biochemistry, maths, engineering, ...). One long-standing issue is the fragmentation between the pkg-scicomp and debian-science Alioth projects. Other agenda items will be discussed prior to Debconf (and this abstract be updated). Contact mbanck@debian.org if you want to participate as panel member in this event (everybody can participate in the audience).

1.9.2   Debian: The ultimate platform for neuroimaging research

Michael Hanke ⟨michael.hanke@gmail.com⟩, Yaroslav O. Halchenko ⟨yoh@debian.org⟩

Over the past decade the neuroimaging research community has fortunately converged on the open source software development model, with the vast majority of all widely used applications and libraries being available as source

code, and a substantial proportion covered by free software licenses. Today, there exists a large code base for data collection (e.g. behavioral psychological experiments), data analysis (e.g. of functional magnetic resonance imaging; fMRI), and data visualization that is productively used in everyday research activities. A representative list of available tools is provided on the NITRC portal [0].

Initially, most of these tools were mere by-products of actual neuroscience research projects, developed by students, and scholars who are, in general, not trained software engineers. As a result, software development in this field still differs significantly from established good practices in the free and open source software (FOSS) community. Many projects start without a clear deployment or management concept, and due to lack of man-power are later on forced into an *ivory tower development* model – restricting the "supported" environment as an attempt to reduce the required maintenance effort. They try to decouple themselves from ongoing developments and include specific versions of all external dependencies into their source distributions. The result is too often a fork that is no longer updated with bugfixes or enhancements. Many useful projects die because it becomes too expensive to update them.

However, it is hard to blame the respective developers, because the sheer number of existing combinations of operating systems, hardware, and library versions makes it almost impossible to verify that a particular software is working as intended. Restricting the "supported" runtime environment is one approach of making verification efforts feasible. On the other hand, development in the free software community sometimes proceeds at an enormous pace, and many projects do frequent releases with sometimes backward-incompatible changes. Forcing a dependency on an outdated release shifts the burden of maintenance and deployment onto users, as it gets increasingly difficult to have older releases available. On the other hand, continuously updating software to the latest developments is a time-consuming task for which there is no immediate scientific benefit.

I will argue that Debian is the ideal solution to this problem. This talk will provide an overview of current projects and individual efforts within Debian that help researchers to maintain a fully-functional Debian-based environment for neuroscience research with minimal effort (e.g. Debian Med, Debian Science). Moreover, it introduces NeuroDebian [1] – a platform specifically targeting the neuroscience community. I'm going to show several case examples of packaging efforts that illustrate typical problems, such as non-commercial/non-standard licenses, non-existing upstream bugtracker and unavailable version control systems. Furthermore, I will offer some evidence that a substantial migration towards Debian is already under way. Popularity

statistics and personal communication indicates that a growing number of researchers see it as 1) making neuroscience software (e.g. for medical imaging) accessible to a larger user-base, and reducing the required maintenance effort on the user side, 2) improving software quality, and 3) enabling developers to spend more time on developing new scientifically relevant features. The talk will conclude with thoughts on what Debian could do to further facilitate its adoption as the ultimate platform for scientific research, such as tailored guidelines for deployment and project management, possibility to run complete regression test suites, as well as reliable usage statistics to justify funding project funding.

[0] http://www.nitrc.org [1] http://neuro.debian.net

1.9.3 Mathematical Software in Debian

David Bremner ⟨david@tethera.net⟩

An open discussion about how to make Debian a better system for mathematicians. Mathematical software is software that is useful to people doing mathematics. Examples include: computer algebra packages; interactive geometry software (e.g. kig); specialized software for e.g. graph isomorphism. For much of mathematics, exact or verifiable computation is an important feature, so packages like GMP are crucial. Natural questions to discuss include what we are doing well, what we could do better, and how could we cooperate more effectively both with each other and with mathematician upstream developers.

1.9.4 New developments in Science Packaging

Michael Banck ⟨mbanck@debian.org⟩

This event will compile short presentations of new developments in basic scientific libraries/packages, or major changes in them.

Please contact mbanck@debian.org if you want to take part in this event.

1.9.5 Overall presentation of the Debian Science

Sylvestre Ledru ⟨sylvestre@debian.org⟩

Debian is a reference in the scientific world. This distribution provides a wide range of scientific software and many fields. While some specific teams existed (Debichem, Deb Med, pkg-scicomp...), there was no generic Science

Team. However, helped with Debian Pure Blend, a general Debian Science team started almost two years ago. The goal of this presentation is to present the team and its past and ongoing actions and its work on various subjects like MPI management or the improvement of the linear algebra libraries system.