

# Setting up a decent dev box for OSS work

[mark.gross@intel.com](mailto:mark.gross@intel.com)  
LPC Student day 2008



# Goals for this talk

- Show one way of setting up a usable developer system for OSS work
- Remove hesitation of participation in OSS work
- Provide a set of “things good to know” when starting out in OSS work.
- talk about strategies for getting involved in a new OSS community
- Kill time on OSS work for fun and profit
- Have a little fun

# System options

- There are a lot of boot options, you'll have to choose
  - Dual boot (windows or mac) and Linux
  - QEMU or Virtualized install (KVM, VMwhere, VPC, Parrales...)
  - Live CD's
  - USB key boot image
- Disk space needs
  - depends on you and your hardware but 20Gig will be plenty to start off with
- System performance
  - can be very productive and happy using last year's laptop models < 600\$
  - 10 min kernel builds are nice and available on bear bones quad core desktop boxen for ~1000\$ wide screen included
- Start with hardware having good Linux device support!



# Base Ubuntu Install options

- Option 1. Basic 8.04.1 desktop install
  - Uses up 2200MB.
- Option 2. Alternate 8.04.1 command line only install
  - storage challenged systems
  - without desktop clutter
  - From the CD boot loader select F3 and choose command line only install
  - Uses up 645MB.

# Adding packages needed for OSS work

- <http://www.thegnar.org/devbox>
- Get the Distro versions of the projects you are interested in.
  - Apt-get build-dep is **magic**
  - Apt-get install is cool too.
- Expect to grow into using upstream versions of your OSS project of choice and possibly uninstalling the distro version.

# Apt-get examples from my website

- #kernel development

```
sudo apt-get build-dep linux-generic linux-image-generic  
linux-doc
```

```
sudo apt-get install linux-kernel-devel kernel-package  
libncurses-dev linux-source docbook-utils xmlto
```

## #xorg development

```
sudo apt-get build-dep xorg x11-apps xserver-xorg
```

```
sudo apt-get install xorg-dev libssl-dev
```

# When build-dep misses

- Google
- Use Synaptic Package manager to search for the missing package.
- Don't panic!
  - ignore dependency conflicts at your own risk

# Additional apt-get friends

- Use `dpkg -l`
- `dpkg` and `apt-get` man pages should be read.
- `apt-get clean`
- `apt-get purge <name>` will remove named package



# Email

- Based on [www.andrews-corner.org/mutt.html](http://www.andrews-corner.org/mutt.html)
- Critical part of OSS work
- Your tools need to scale with high volume mailing lists
- Tools should make list mailing list behavioral norms natural (no-html, proper line wrapping, bottom posting...)

# Tools I use

- Mutt : [www.mutt.org](http://www.mutt.org)
- Procmail : [www.procmail.org](http://www.procmail.org)
- Msmtp [msmtp.sourceforge.net](http://msmtp.sourceforge.net)
- Fetchmail : [fetchmail.berlios.de](http://fetchmail.berlios.de)
- “working” config file examples on my web site  
[www.thegnar.org/devbox/email.html](http://www.thegnar.org/devbox/email.html)
- 12 extra - extra boring - slides in OSCON version of talk: [www.thegnar.org/devbox/oscon2008.pdf](http://www.thegnar.org/devbox/oscon2008.pdf)

# Strange names

- Mail Transfer Agent (MTA)
  - Mail Retrieval Agent (MRA) -- fetchmail
  - Mail Sending Agent (MSA) -- msmtmp/ssmpt
- Mail Delivery Agent (MDA) -- procmail
- Mail User Agent (MUA) -- mutt

# mutt

- mutt is old school
  - [www.mutt.org](http://www.mutt.org)
- “alias” file set up in the .muttrc file.
  - alias dorkbotpdx  
dorkbotpdx-blabber@dorkbot.org
  - alias AHRS ahrs user1@isp1.com,  
user2@isp2.org, ...

# SCM related



# Other SCM's you should know a little about

- SVN
  - CVS
- Mercurial (Hg)



# GIT

- Git is a distributed version control system.
- hard to wrap my head around
- many tutorials exist.
  - you should work through a few of them.
  - none of the tutorials cover : **git-remote**

# common git commands

- git-remote :

- `git-remote add linus git://git.kernel.org/.../torvalds/linux-2.6.git`
- `git-remote add fboot git://git.kernel.org/.../arjan/linux-2.6-fastboot.git`
- `git-remote add internal mgross@linux-repo.jf.intel.com:/internal/kernel`

- git-status

- git-reset -hard

- git-clean -d

- git-merge



# more common git commands

- git-rebase
- git-branch -a
  - git-branch fastboot fboot/master
- git-fetch
  - **git-fetch fboot**
- git-checkout
  - git-checkout -b fastboot fboot/master

# Kernel related

- You must read:
  - `./Documentation/{SubmitChecklist,  
SubmittingDrivers, SumbittingPatches}`

# tools

- Make help
- Ctags : make tags
  - `gvim` works well with the TAGS file
- CSCOPE : old school code browser
  - `make cscope`
  - `cscope -kd`
  - To exit `cscope` you hit `^d`
- Make {htmldocs, mandocs,...} output are interesting.

# ./scripts/checkpatch

- It's a format and style checking tool that codifies the Linux kernel coding standards
- “./scripts/checkpatch <pathname of patch>”  
from within the kernel root tree.

# Utilities



# Using quilt

- Quilt is awesome!
- Many folks use quilt within a git tree.
  - use git to track upstream kernels
  - use quilt to keep track of their work
- Use:
  - `quilt new my_cool_feature.diff`
  - `quilt add <existing file>`
  - `quilt refresh`
- `Quilt edit <filename>` is a good way to add file before editing.

# Using quilt

- quilt -h
- quilt pop
- quilt push
- quilt series
- quilt refresh
- Gotchas
  - parent directories with a directory named “patches”
  - forgetting to add files

# git with quilt

- checkout your code base with git
- create your new working patch with quilt
  - add files and work work work
- quilt refresh then quilt pop all your patches
- git-fetch
- git-merge
- quilt push your work and fix up any issues you may have.
- When ready or just post the patch to LKML.



# Web and communications



# minicom

- (the?) serial terminal program.
- very old school
- ^a^z gets a help menu of sorts
- some distros give it permission issues with the /dev/tty\* devices
- minicom -s

- clients:
  - xchat seems to be the easiest for me to use
  - irssi
- Don't post large traces, like a failed compile error logs, to IRC.
  - <http://rafb.net>
- naming a person in your text will “beep” on their remote.

# More IRC

- type-oh correction is normally expressed in regex:
  - `s/<your-type-oh>/<what you wanted it to say>/`
  - it won't redact your screw up
- / commands are how IRC works.
  - `/msg <nick> <message text>`
  - `/kick`
  - `/me`

# Simple web site updates

- ncftp is the nicest FTP client I know
  - `ncftp -u <loginname> IP/site URL`
- Use ssh based file transfers if you can e.g. scp, sftp
- Seamonkey (mozilla) Composer.
  - its dumb, but works
  - It has some warts (save work often)

# Python stuff



# Python must haves (for me)

- code checkers : pylint, pyflakes
- python shell : ipython
- python docs
- graphing tools : matplotlib
- scientific stuff : scipy, numpy
- serial port tools : python-serial.
- image processing tools : python-imaging

# iPython is cool

- ? will provide basic help
- help() will give access to all of the python help on your system.
- See Jeff Rush's showmedo videos on iPython
- <http://showmedo.com/videos/series?name=C>



# Strategies for getting into a new OSS community

- Learn about the community and technologies in stealth mode.
  - lurk on mailing lists and IRC channels.
  - Know your OSS community's governance and mailing list policies.
- Users group participation
- Go to code sprints!
- Start making a name for yourself
  - File bug reports
  - Fix a bug or three
  - Test, and report issues, even minor ones.
  - Do something nontrivial with technology.
  - Give a talk at a user group or conference
    - Hang out with cool folks, and have fun

# OSS work for fun and profit

- Use the OSS technology for something cool.
  - Ice breaker: Ask about what's new and cool when getting together with more active community members.
- Know that employers will scrape OSS mailing lists and IRC channel logs for your name if applying for a job.
- Don't be intimidated by the rock stars.
- Make an effort to not be “annoying”
- Make it your goal to be helpful to the project and you'll be accepted quickly
- If you actually do help then you quickly discover you too can be a “rock star” if you want to go for it.
  - It just takes time and effort.

# Some resources

- [www.mutt.org](http://www.mutt.org)
- [www.procmail.org](http://www.procmail.org)
- [msmtp.sourceforge.net](http://msmtp.sourceforge.net)
- [fetchmail.berlios.de](http://fetchmail.berlios.de)
- [git.or.cz](http://git.or.cz)
- [savannah.nongnu.org/projects/quilt](http://savannah.nongnu.org/projects/quilt)
- [www.thegnar.org/devbox](http://www.thegnar.org/devbox)
- [www.showmedo.com](http://www.showmedo.com)
- <http://live.gnome.org/GnomeLove>

# THANK YOU



Software and Solutions Group

