



The birth

Before the development-fest kicks off, Ubuntu hackers need to decide which features to include.

Imagine yourself as an Ubuntu developer: you've just got the latest, shiny, brand-new release out the door, and Linux users around the world are eagerly installing it. You've grafted away on writing code and fixing bugs, ready for a well-deserved rest – but no! It's already time to get cracking on the next release. Ubuntu follows a six-month release schedule (having deviated once in 6.06 for stability reasons), so the turnaround for getting new features implemented is very tight.

The Ubuntu development team comprises paid and voluntary coders, the former working full-time on the distro and employed by Canonical, while the latter chip in whenever they can. Despite this marriage of paid and unpaid work, there's virtually zero friction between the two groups – especially as Canonical has employed some Debian developers in the past. For many coders, part-time Ubuntu hacking provides a possible career-path towards salaried distro making.

Once a new Ubuntu release has been sent out to the mirrors, the process of choosing features for the next release gets underway. Matt Zimmerman, Ubuntu's CTO and Technical Board chairman, explains the process: "Ideas are suggested by the Ubuntu community and by the developers themselves. If a developer is interested in the idea, they flesh it out into a technical specification, which can then be the basis for a development project."

These feature ideas are assembled in Launchpad, Canonical's developer support site – for Gutsy, they're listed at <https://launchpad.net/ubuntu/gutsy/+specs>. Some of the feature ideas are discussed face-to-face amongst developers, as Zimmerman continues: "Much of the activity happens at the Ubuntu Developer Summit, near the beginning of each release cycle. These specifications are reviewed by a team of technical reviewers to assess their correctness and feasibility."

Priorities, priorities!

Each new feature idea is given a priority (high, medium and low) dependent on how much in demand it is, what effects it will have on the distro's overall stability and whether it can feasibly be accomplished in time. Launchpad also provides constant updates on the process of a new feature: developers can see if it's just being started, making good progress or ready for beta testing. They can also see who is working on each project.

But how is the development effort split up? Zimmerman enthuses: "Canonical itself determines which projects its developers engage in, while community developers make their

Rising through the ranks

There are two types of Ubuntu developer: MOTU and Core. The first forms part of the 'Masters of the Universe' team, a large group of hackers who maintain software in the distro's Universe and Multiverse software repositories. Universe encompasses packages which aren't deemed important enough for the main base system, but which users may want nonetheless. New developers join the MOTU team, packaging programs in Ubuntu .debs and maintaining them for bug and security fixes. After a developer has proved

My favourite Gutsy feature: Mark Shuttleworth



"That would have to be a toss-up between Compiz, which gives us a wonderful multi-dimensional desktop experience, and the new Mobile edition of Ubuntu which is designed for internet tablets that let you carry the net in your pocket."

own decisions about how to spend their time. In practice, many of the projects that Canonical developers pursue are also their own ideas, but we also make some top-down decisions." So voluntary developers can cherry-pick their own favourite features to hack on, whereas Canonical adds more direction to its paid team.

Feature proposals come from all aspects of the Linux world: some are additions found in another distros, some are nascent technologies creeping out of a random SourceForge site, and some are simply cool gadgets that may be useful in the next Ubuntu release. Once the feature list is complete, the Development Manager (for Gutsy, Scott James Remnant) sends out a summary to the development mailing list (<http://tinyurl.com/2cxfev>), and the coding begins...

› <http://launchpad.net> helps developers to organise and prioritise Gutsy's new features.

capable of producing good Universe packages, s/he can apply to the Ubuntu Technical Board and Community Council for promotion into the Core team. If the developer has done good work: reliable packages, strong communication skills and broad Linux knowledge, s/he may be able to join the Core team, working on critical packages in Main and Restricted repositories (ie many of those shipped on Ubuntu discs). Ubuntu has a code of conduct to ensure that developers help, respect and collaborate with one another.

Coding time

"Making a distro is 1 per cent inspiration, 99 per cent perspiration" – as Thomas Edison would have said if he lived in these Linux times...

Although Gutsy is in the midst of a huge development charge right now, the overall goals for the release have been set, as described previously. These are the key features and changes that the Ubuntu team hopes to include in 7.10 – but as always with a distro release, some may be dropped if they can't be made functional or stable enough by the release date!

» **Desktop Gutsy** will ship with Gnome 2.20, and its Kubuntu variant will have KDE 3.5.7. Those looking to get a glimpse of KDE 4 will be able to install a release candidate alongside the stable version. Gutsy will be the first release to include *Compiz Fusion*, providing swish 3D desktop effects on supported graphics chips. Users will be able to switch back to vanilla window managers with a few clicks. Also, a new GTK-based graphical config tool for screen settings (resolution, colours etc) is in the works.

» **X.org** X, the display system, will undergo a few changes: version 7.3 will be included, providing hot-plugging of display devices. One of the goals here is 'bullet-proof X' – that is, an X setup that safely falls back to standard video modes when there's a problem, instead of collapsing in a heap on the command line. For the sake of all Linux newbies in the world, we're cheering for this feature here at *LXF Towers*!

» **WinModems** Yes, those nasty software-driven modems, most commonly found on laptops or given away free with cheap broadband connection packages, will be supported. Well, some of them. Gutsy will provide access to some 'restricted' (ie not completely free or open source) drivers to help get people online.

» **Security** AppArmor, a security framework that lets admins lock-down programs and the resources they can use, will be available as an optional extra. This is the system used by Novell in SUSE; Red Hat opts for SELinux instead.

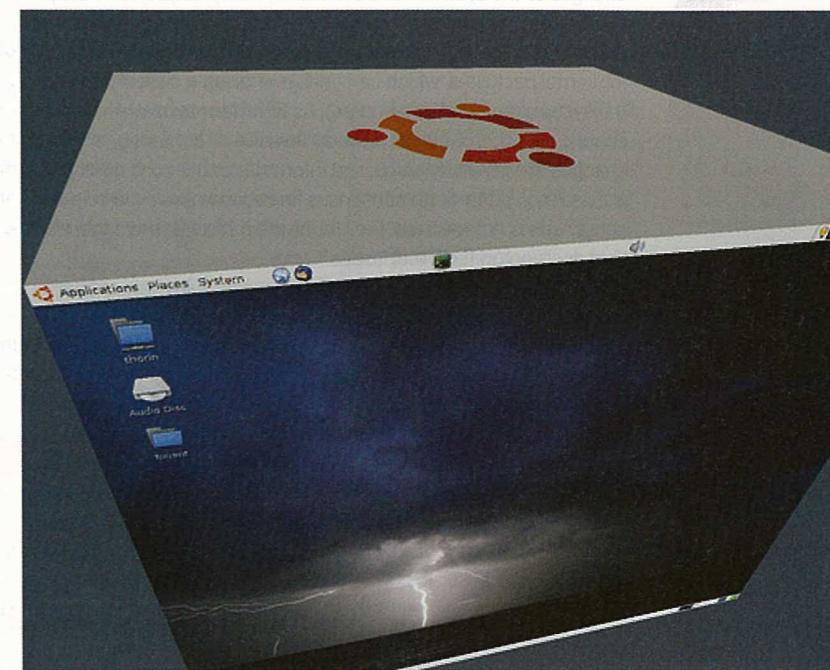
» **Free Flash** Some Ubuntu coders are investigating the possibility of a free Flash player being included. Currently, the most compatible player is the non-free one from Adobe, but the Ubuntu team is considering *Gnash*, a free software alternative. *Gnash* still lags behind the proprietary player, though.

» **Free as in freedom** A technical review is underway on the possibility of a non-restricted Gutsy installation – that is, the ability to install from the CD without copying any non-free components to your hard drive.

Some believe this is a more sensible approach than having a completely separate distro for those who want pure open source/free software goodness. This will be called Gobuntu.

» **Flavours** A new variant, Ubuntu Mobile and Embedded Edition, will be available for developers working on internet tablets and other small-scale devices. Also, the Ubuntu Server project will have a new team beavering away at it at Canonical.

So, these are the current feature goals for Gutsy, and by the current rate of development, it looks like most of them will be in place for the final release. At this



Behold! 3D desktop effects are on the way, thanks to the integration of *Compiz Fusion*.

stage, a few nerves start twitching in the lead developers – after all, news websites and magazine articles (like the one you're reading!) start discussing the new features, so they want to ensure that nothing major is dropped and nobody is disappointed when the final release is cut. Zimmerman describes how this problem intensifies as development ramps up: "The challenge becomes keeping abreast of many parallel projects and their progress, so that we can deliver what we planned on time."

Code frenzy

And the coding begins. Packagers from the Core and MOTU teams start grabbing the latest source code of thousands of programs, adding the .deb build scripts and generating Ubuntu-flavoured packages. Here, the distro is like water, constantly flowing and changing underneath the developers, and as critical dependencies

Key Ubuntu people

Some names of repute in the Ubuntu hacking team:

» **Mark Shuttleworth** Uber-wealthy one-time-astronaut, made his fortune by founding Thawte, and now the self-appointed "Benevolent Dictator For Life". He sets the overall direction of the project and, as a former Debian developer, chips in with some technical decisions too.

» **Matt Zimmerman** Chief Technical Officer. Zimmerman looks after the coding side of the

distro, making sure the development and release processes aren't hindered by major bugs, developer arguments or whimsical feature requests.

» **Brian Murray** Chief bugmaster and manager of the Ubuntu QA team, responsible for squashing as many bugs in the distro as possible before a release.

» **Corey Burger** Head of the documentation project, which helps to make the distro more accessible and provide translations for a wide range of languages.

sure that the distribution exudes positive vibes.

» **Troy Sobotka** Artwork group lead. His team works on themes and icons to keep Ubuntu looking fresh (well, as fresh as a brown theme can look!) and ensuring that the distribution is visually consistent throughout.

» **Matthew East** Oversees the documentation project, which helps to make the distro more accessible and provide translations for a wide range of languages.