

The 64 Studio distribution – creative and native

Daniel James
64 Studio Ltd.
PO Box 37
Freshwater,
PO40 9ZR,
Great Britain
daniel@64studio.com

Free Ekanayaka
64 Studio Ltd.
PO Box 37
Freshwater,
PO40 9ZR,
Great Britain
free@64studio.com

Contents

1. The 64-bit question
2. Package selection
3. Why Debian?
4. The business model
5. Challenges
6. Conclusion
7. Acknowledgements
8. References
9. Appendix: Packages in version 0.6.0

Abstract

This paper describes the high integration of proprietary software for the creative desktop, and the effort involved in creating a free software alternative which will run natively on the latest 64-bit x86 hardware. It outlines the author's reasons for creating a 64-bit distribution based on Debian, the packages selected, the business model of the 64 Studio company and the challenges for future development.

Keywords

64-bit, Debian, content creation

Introduction

If we take a step back from pure audio software for a moment, and look at the state of creative desktop computing tools more generally, it's obvious that there has been a lot of consolidation among the proprietary software vendors in the last couple of years. For example Yamaha swallowed up Steinberg, Adobe bought Syntrillium (the creators of Cool Edit), Avid bought Digidesign and Apple bought Logic. Adobe's 'partnership' with Macromedia became a takeover, and now the

company positions its extensive range of multimedia applications as the 'Adobe Platform'. What this means is that regardless of the hardware or operating system in use, the mainstream creative desktop of the near future is likely to represent a highly integrated set of non-free applications from a very small number of vendors. We can expect these proprietary applications to be well tested for performance, reliability and usability.

We believe that it will be very difficult for GNU, Linux and other free software to compete for users on the multimedia desktop unless it can achieve a similar level of integration and polish. Without a significant user base, it becomes difficult for free software to maintain the hardware support that it needs. Reports indicate that it is becoming progressively more difficult to obtain full specifications for video card driver development, and several of the most popular high-end audio interfaces, particularly the FireWire models not running BeBoB, remain without the prospect of a free software driver. We aim to deliver a viable and sustainable creative platform based on free software, and partner with hardware manufacturers to ensure the availability of fully-supported components and peripherals.

1 The 64-bit question

Since any software project takes a while to get to a mature stage, when we launched a new multimedia distribution last year, we decided to concentrate on the kind of desktop systems which we believe will be common among creative users in the future.

We're interested in 64-bit x86 for two main reasons - the improvements in memory architecture, allowing commodity machines to have many gigabytes of RAM, and the opportunity to drop support for legacy PC hardware. From the point of view of a distribution, any technology that narrows down the field of potential hardware combinations is a great advantage. We don't have to support ISA bus sound cards and we don't have to care if the binaries won't run on a 486.

That may sound a little harsh for owners of older hardware, but there will be plenty of 32-bit GNU/Linux distributions around for some time, and the relentless downward spiral in the cost of newer technologies looks set to make using any hardware older than a year or two quite counter-productive. For example, the HP laptop which we are giving this presentation on is an entry-level model from a department store in the UK. It has a 1.6GHz AMD Turion 64-bit processor and 1GB RAM as standard. It cost less than a far slower generic white-box PC of a couple of years ago, and it probably uses a great deal less energy too.

We've had native 64-bit Linux on the Alpha and the Itanium for years, but these architectures never reached the mainstream desktop. SGI has an Itanium2 based GNU/Linux desktop product aimed at the creative market, but it costs US \$20,000 per machine. Compared to Windows or any other operating system, GNU/Linux clearly had a head start on x86_64, and you can choose from a range of natively compiled desktop distributions for the hardware. Unfortunately for the creative user, all of these are aimed at the general purpose computing audience. It's impossible to be all things to all people, and what's good for the so-called 'consumer' is rarely right for the content creator.

2 Package selection

For example, typical distributions use Arts or ESD to share the sound card between applications, while most GNU/Linux musicians would want to use JACK - admittedly more complex, but far more powerful. I (Daniel) was once asked what was so difficult about JACK that means it isn't

found as the primary sound server in any mainstream GNU/Linux distribution. I don't think it is difficult to use, but for the time being it still requires a patched kernel, and some knowledge of sample rates and buffers. Many non-musical users just want to be able to throw audio at any sample rate to the sound card, and could care less about real-time priority.

In addition, the creative user's default selection of applications would be very different to - for example - a sys-admin. Even gigantic distributions like Debian don't package all of the specialist tools needed for media creation, and the integration between packages is often less than perfect. So the goal of 64 Studio is to create a native x86_64 distribution with a carefully selected set of creative tools and as much integration between them as possible.

Today, we have free software applications covering many of the creative disciplines other than audio or music, including 2D and 3D graphics, video, and publishing for the web or print. Unfortunately media creation, when compared with media 'consumption', remains a niche activity, even on Linux. This niche status is reflected in the fact that none of the mainstream Linux distributions work particularly well 'out of the box' for media creation - but to be fair, Windows XP or OS X also require many additional packages to be installed before their users can realise the full creative potential of their chosen platform.

Of course specialist Linux audio distributions already exist, including AGNULA/DeMuDi, Planet CCRMA, dyne:bolic and Studio to Go!, with a good level of integration for music-making. But all of these other audio distributions are x86 only so far, and there are few specialist distributions in the other creative fields. Ratatouille, a Knoppix-based distribution designed for animators, is one exception.

Switching to native 64-bit software doesn't necessarily realise an instant and obvious improvement in performance on the same hardware, but we think that if we create a native platform, then application developers can begin to realise the benefits of 64-bit processor

optimisation and an improved memory architecture. Even in the short term, it makes more sense than building i386 binaries.

But there's a problem with specialist distributions. Since they have relatively few users, they usually end up being maintained by a single person. External project funding, whether from the state or a venture capitalist, is often unreliable in the long term, and can steer the agenda of the distribution away from that of the users.

Since we believe maintaining a niche distribution is simply too much work for a volunteer to be expected to do, we set up a company to pay developers to create and maintain the system using the Custom Debian Distribution framework. You may know of Free's work on CDD from recent releases of the AGNULA/DeMuDi distribution. Most of the packages in 64 Studio come from the Pure 64 port of Debian testing, with some from Ubuntu, some from DeMuDi and some custom built.

3 Why Debian?

A more obvious choice might be Red Hat, given that many of the high end (which is to say expensive) proprietary tools used in Hollywood studios and elsewhere are sold as binary-only Red Hat packages. However, the split between Red Hat Enterprise and Fedora Core presents serious problems for any derived distribution. On the one hand, you could rebuild Red Hat Enterprise from source as long as you removed all Red Hat trademarks, but that's a lot of extra work - and you'd have to follow Red Hat's agenda for their distribution, which you couldn't have any input to. We doubt that you'd get much goodwill from Red Hat for 'improving' their distribution either.

On the other hand, you could build a distribution on top of Fedora Core. It's broadly Red Hat compatible, and there are the beginnings of a community process taking place - although it's still far more centrally controlled than genuine grass-roots distributions. The key problem with this approach is that Fedora Core is not designed or built to actually be used. We can say this with

some confidence because I (Daniel) was able to ask Michael Tiemann, former Red Hat CTO and now vice president of open source, this question myself. Fedora Core remains a technology preview for Red Hat Enterprise, and the Fedora Project has absolutely no commitment to stability or usability. If Red Hat wants to try a major update to see what breaks, it can.

Debian does have a commitment to stability, and a bona-fide community process. There are other reasons for favouring Debian over Red Hat, not least of which is the long-established support in Debian for seamless upgrades with apt-get, since on the creative desktop we'll be upgrading continuously. The work of the Debian Pure 64 port team is of a very high quality, not to mention that of all the many Debian package maintainers.

We recognise that whatever packages we put into 64 Studio, users will want some of the packages that we haven't included - so being able to use thousands of binaries straight from the Pure 64 port without modification is a major advantage. Because we're sticking very closely to Debian with the 64 Studio design, users can install any application from Pure 64 simply by enabling an additional apt source. This includes most of the well-known applications with the exception of OpenOffice.org, which just won't build natively on x86_64 yet.

In fact, 64 Studio is not so much a distribution based on Debian as a Debian remix. Free is in the process of becoming a Debian Developer, so we will be able to contribute our improvements back directly - where they are Debian Free Software Guidelines compliant. However, we do benefit from the flexibility of not being an official part of Debian. For example, the Debian project has decided that it does not want to package binary audio interface firmware, which is required to be loaded by the driver for the interface to work. That's fair enough, and we understand the reasons for their decision, but it's a major problem if you own that kind of interface, because it won't work out of the box.

This kind of hardware - effectively reprogrammable on the fly with a new firmware blob - is only going to become more common, and

not just for audio interfaces. So for the sake of our users, we have to support it. Otherwise, free software will become significantly harder to use than proprietary equivalents - and that's not a future we want to see. As end users, we couldn't modify our sound cards when they were pure hardware, so we don't think it's any worse that they now require a binary upload. At least now there is the possibility of creating our own firmware and uploading that instead, which we didn't have before.

Our first alpha release was based on Fluxbox, because this window manager places minimal demands on system resources, and is very quick to learn, since there isn't much to it. However, we have since switched to a stripped-down Gnome install. Again, this is because we don't want to make free software too difficult for people who are used to proprietary tools. This doesn't mean that we will dumb down the interface or clone the look of other platforms, but - for example - we can expect new users to assume that the program launching menu is in the lower left corner of the screen. There are also expectations about drag and drop file management, or GUI-based system configuration tools. Fluxbox is very fast, but it's an extra thing to get used to on top of everything else.

4 The business model

Since we want to pay developers to work on 64 Studio, part of making the distribution sustainable is creating a viable business model based on free software. The maintainers of the 64 Studio distribution are fundamentally in an editorial role, selecting the most appropriate software from the many thousands of packages available, and putting it into a convenient snapshot. Since the software is free software, it would be churlish of us to demand that people pay us to do this, but if we provide something of value then it should be worth a modest (and completely optional) subscription. We believe Red Hat's compulsory subscription model has cost its Enterprise distribution a lot of potential users. Apart from being ethically questionable in the context of software contributed to the distribution at zero cost, as a systems manager at a well-known studio with hundreds of Red Hat

desktops put it, "Why should we have to pay for support every year whether we need it or not?"

Community support often meets or exceeds the quality that proprietary software vendors provide, but people tell us that it's reassuring to have some paid-for support available as an option. Sometimes our questions are just too ordinary to interest people on a mailing list or forum, or at the other end of the scale they can require patience and time-consuming research to answer. It can sometimes be difficult to get the help you need when you're up against a project deadline. We believe that by covering one kind of desktop user really well, we can provide detailed support for the people that need it at a reasonable cost. For the people that don't need support, or are planning large deployments where per-seat licences would be prohibitive, it's still free software - and we're not going to lock people into support contracts in order for them to access updates either.

We also offer the 64 Studio codebase as a development platform for OEMs building multimedia products on x86_64 hardware. We believe this enables these companies to reduce their development costs and time-to-market. We are considering producing a server edition of the distribution in future that would combine a fast and simple install with pre-configured services, so that a workgroup file server or a streaming media server could be set up in a few minutes - and these services would work right away with 64 Studio desktop machines of course. In the longer term, we hope that 64 Studio will go beyond packaging and integration work to contribute directly to application development, particularly where 'missing links' are identified.

5 Challenges

There are a number of challenges we still have to face. The first is following the rapid pace of kernel development. In version 0.6.0 we were using Linux 2.6.13 with Ingo Molnar's real-time pre-emption code and a few other patches. At one time these patches didn't build on x86_64 at all, and as far as we knew, we were the only native 64-bit distribution using them at the time. The

indications from our beta testing community are that this combination works really well for audio with full pre-emption enabled, the most aggressive setting. For the time being we are using the realtime-lsm framework to give real-time priorities to non-root users, because we know it works. We may switch to rlimits in the future, as this code has been merged into the mainline kernel for some time now.

Another challenge we have to deal with is the Debian community process. We are not in a position to demand anything from the Debian developers, we can only suggest and encourage. If there's a real roadblock within Debian, we have the option to create a custom package, but obviously that's something we'd rather not do.

A third challenge is the issue of support for proprietary formats within free software. At the level of encoding and decoding, we think the best solution we've seen is the GStreamer plugin collection, which as far as we can tell meets the requirements of free software licences regarding linking, and also the legal requirements of the patent holders. It's simply not sustainable to expect users to locate and download libraries of dubious legal status, and install these by themselves. Apart from any ethical problems, it's impossible to support users properly in that situation. In addition, using these libraries is likely to be out of the question for an institutional user, such as a college.

At the level of project interchange, for example moving a complex project from Ardour to ProTools, there does seem to be a move among proprietary audio applications towards support for AAF, the Advanced Authoring Format. Free software must support this kind of high-level project compatibility format, otherwise it doesn't stand a chance of gaining a significant user base in this area. When we talk to people in the music industry, it's almost a mantra that 'everyone mixes in ProTools'. We're not aware of any free software audio application that supports ProTools format import or export directly, but at least with AAF we have the chance of finding a middle way.

6 Conclusion

64 Studio is available for download as an .iso image, and the distribution is seamlessly upgradeable with apt-get of course. We'd be more than pleased to hear your test reports and suggestions for the distribution - you can help us make free software the creative desktop of choice.

7 Acknowledgements

We would like to thank all the free software developers who make building a distribution like 64 Studio possible.

8 References

64 Studio homepage
<http://64studio.com/>

The 64-bit x86 architecture
<http://en.wikipedia.org/wiki/AMD64>

Debian Pure 64 port
<http://amd64.debian.net/>

Advanced Authoring Format
<http://aaf.sourceforge.net/>

9 Appendix

Some of the packages included in 64 Studio release 0.6.0:

CD

sound-juicer
cdrdao
dvd+rw-tools
gcdmaster

Graphics

gimp
inkscape
blender
gphoto2
gtkam
gtkam-gimp
gthumb
yafray
dia
libwmf-bin
ktoon
pstoedit
sketch
imagemagick
perlmagick
xsane

Internet

gftp
bluefish
linphone
gaim
gnomemeeting

JACK

jackeq
jack-rack
jamin
meterbridge
qjackctl

Audio

alsa-base
alsa-firmware
alsa-source
alsa-tools
alsa-tools-gui
alsa-utils
flac
speex
swh-plugins
tagtool
tap-plugins

vorbis-tools
totem-gstreamer

Base

bittornado-gui
bittorrent
gnome-system-tools
ia32-libs
nautilus-cd-burner
vorbis-tools
vorbisgain

Office

abiword-gnome
abiword-help
abiword-plugins
abiword-plugins-gnome
gnumeric
gnumeric-doc
gnumeric-plugins-extra

Publishing

scribus
evince

Notation

notedit

Recording

ardour-gtk
ardour-session-exchange
audacity
timemachine

Sequencing

hydrogen
rosegarden4
muse
seq24

Synthesis

ams
amsynth
linuxsampler
qsampler
qsynth
vkeybd

Video

kino
libtheora0
dirac

Kernel

kernel-image-2.6.13-1-multimedia-amd64-generic
realtime-lsm