These are the comments to Ruario's blog post, copied from the archive.org page:

Audrius Kažukauskasaudriusk # Tuesday, September 27, 2011 10:11:23 AM

Another thing that people using some other distros often don't know is that because Slackware doesn't manage dependencies, there's no need to split packages into -devel, -doc, -utils, etc. You get everything (development headers, documentation, …) inside a single package (as often intended by authors of particular software). This greatly reduces the number of packages and further simplifies things.

Ruarí Ødegaardruario # Tuesday, September 27, 2011 10:44:29 AM

Indeed, this means that whilst the official Slackware repository is smaller than many other distros, it is usually easier on Slackware to use a source package directly from the author, i.e. “./configure && make && make install” is far more likely to work right out of the box.

Handy for testing those interesting little applications one stumbles across from time to time, that don't seem to be in any repositories.

Ruarí Ødegaardruario # Tuesday, September 27, 2011 10:54:11 AM

That said, hassles with source packages tend to be bigger on the so called 'newbie friendly' distros like Ubuntu and derivatives. And this I suppose is fair because the maintainers of those distros don't expect regular users to frequently compile from source. The other distros where the users are expected to regularly compile from source (e.g. Gentoo, Arch, etc.), don't tend to have such problems either. 😉

Ruarí Ødegaardruario # Tuesday, September 27, 2011 11:25:23 AM

I should probably also add a little extra comment regarding doing a 'full install'. Whilst this is recommended and generally has few real world downsides, I am not trying to imply that: it is the only option, that every Slacker does this, or that there is never a good reason to do a slimmer install. The installer allows you to completely customise your install, many Slackers make use of it and there are certainly valid reasons for doing this. An example being if you were installing on exotic hardware where disk space cannot easily be added (e.g. some of the small ARM-based systems) but this is far from the only reason. Others such as 'to see what is possible', are equally valid.

All I'm really trying to get across is that in the majority of causes a 'full install' is a viable option on Slackware and shouldn't be immediately discarded as a possibility, as you might do with other distros. I have often seen comments on forums from users who are new to Slackware, where they try and start with something minimal because they think they should, rather than because they have a genuine need. It is a shame if they then encounter problems because it really doesn't need to be that hard!

Even if you do plan on performing a striped down installation, if you are new to Slackware, I still think a 'full install' is a good place to start. It allows you to play around with removing packages and seeing what happens. This gives you a chance to build up knowledge of how things work together and is a lot more sensible than attempting to do a nice slim 300Mb install at your very first attempt. Also, one good thing about building up knowledge slowly rather than doing a rush job, is that what you learn is more likely to stick.

Finally, it is worth remembering that whilst a slim install is undeniably harder than many other distros the very first time you attempt it, each subsequent small install will get easier and faster as you learn the interrelationships between packages. Once you have gained this knowledge, a small install can be done relatively quickly, especially if automated with tag files.

Ruarí Ødegaardruario # Tuesday, September 27, 2011 12:21:24 PM

It may sound obvious but my personal tip for those looking to slim down the install would be to start by considering the big packages that you understand, rather than trying to take out a few tiny utilities and drivers (that may only be a few Kb each) or by removing libraries that you have never heard of (but could turn out to be crucial).

For example, you could consider dropping the whole KDE series if you aren't a KDE fan to save around 900Mb. Another good option might be the 415Mb (installed size) kernel-source package. Whilst you will need this package if you want to compile a custom kernel or if you need to compile special kernel modules (like the VitualBox Guest additions), it can be a nice saving if you don't need it. Even if you do need it you can still consider removing after you are finished with it, as it is the type of package you typically only want at the beginning, when first setting up your system.

You can easily take the install size down to less than half by just removing a few of those big well known packages. And remember it is the packages' install size you care about. kernel-source is only 62Mb compressed, so you might not have considered it if you were only looking at the size of the compressed package.

I'd also personally favour removing as little as you can get away with, so that you can continue to enjoy the benefits of having a nice varied collection of solid packages to hand should you ever need them.

For other view points and tips, a search for 'minimal slackware' in your search engine of choice is likely to lead you to some useful resources.

slackwrdave # Tuesday, September 27, 2011 10:05:10 PM

I thoroughly enjoyed this post and comments. Nice job.

Manpreet Singh DhillonManiDhillon # Tuesday, September 27, 2011 11:32:08 PM

Nice article. You thoroughly discussed what most of the Linux users are concerned about in

Slackware, dependencies. Though in my humble opinion managing and making a minimal working install of Arch is quite fast. Though Arch and Slackware work on the same philosophy that of being user centric not user friendly. I am not an expert on Linux but I am using different Linux distros since 2006 but still I sometimes got many problems in Slack and thus changed to Arch for same needs.

Corey Mwambacoreymwamba # Wednesday, September 28, 2011 3:42:43 AM

Thank you very much for the pointer to this Ruari!

I actually think I'm in a position to switch now - It's not even as if I'm using the openSUSE kernel any more... and if Slackware just uses tarballs then it removes a layer of complexity.

Time for a back-up, reset the SSD, then install, wish me luck! smile

Mad Scientist (عادل)qlue # Wednesday, September 28, 2011 4:42:26 AM

Now I need a second machine for experimenting on! sherlock.

Ruarí Ødegaardruario # Wednesday, September 28, 2011 7:15:13 AM

@coreymwamba: Good luck! 😬

Also, here are a few tips for you, if you don't mind. 😉

The simple text file documentation provided with the install media is actually really helpful on Slackware and is regularly updated. I would encourage you to actually read README.TXT, Slackware-HOWTO, CHANGES_AND_HINTS.TXT and README.initrd. These files aren't that long and should prevent you from making typical mistakes if you follow them closely.

In addition here are a couple of links you might find useful when you first set off on your Slackware adventure: http://wiki.linuxquestions.org/wiki/Slackware-FAQ http://duganchen.ca/writings/slackware/setup

Also just a few quick tips of my own regarding things that IMHO seem to be brushed over a little too quickly in some of the other documentation and/or guides that I have read.

If you want to know anything about the packages on your system look at the package logs, which tell you version numbers, install and package sizes, package contents, etc. you can read them with less, a text editor or check specific things with grep. The main ones are found in /var/log/packages/. The post install scripts (that were run after a package was installed) are in /var/log/scripts/. Also removed packages have logs in /var/log/removed_packages/ and /var/log/removed_scripts/

Remember that Slackware uses a BSD-style init system. All of Slackware's native startup scripts are in /etc/rc.d/. Typically scripts that are executable are the ones that are run on startup, which means that if you remove the executable bit (i.e. 'chmod -x rc.scriptname'), they won't run on startup. Or add it ('chmod +x rc.scriptname') if you do want them to run. Easy, eh? If you need to start some extra system wide service that you have installed from a third party repository, and it either doesn't provide a startup script in this location or does but is not automatically run once its script is made executable, then add an entry to rc.local to call it (and perhaps one to rc.local_shutdown to have it cleanly stop when you shutdown). That is about it for starting and stoping services on boot.

Also, since Slackware provides very few system wide configuration tools, you are going to want to get familiar with the files in /etc because everything is configured in these files when making system wide changes.

If you are unsure about anything and can't find documentation check out Slackware's official forums, which are actually hosted on Linuxquestions.org. They are a good bunch of knowledgeable people (Patrick Volkerding himself occasionally even answers people) but they do expect you to do your homework so run a search first before you ask! wink
http://www.linuxquestions.org/questions/slackware-14/

Finally, the following Blog is quite interesting to keep up to date with what is going in the Slackware world: http://slackblogs.blogspot.com/

@qlue: You can always test under VirtualBox. 😉

Ruarí Ødegaardruario # Wednesday, September 28, 2011 8:09:13 AM

@coreymwamba: How could I forget. Most important of all you will want op2slk to convert Opera Next tar packages into Slackware format. wink

Or if you prefer I put up pre-converted copies here.

Mad Scientist (عادل)qlue # Wednesday, September 28, 2011 12:33:31 PM

@Ruari I only have a netbook. It's just not capable of virtualisation. 😛 (the Intel Atom is just a glorified 386 masquerading as a 686 after all. 😉

Cutting Spoonhellspork # Thursday, September 29, 2011 1:25:08 AM

Originally posted by qlue:

the Intel Atom is just a glorified 386 masquerading as a 686 after all.

More like a P2 with faster gates/more cache, several forms of SSE and Hyperthreading. There's a special “Atom” config set on several newer development kits, especially regarding multimedia. Yes it's still “just” an Atom, but 4 virtual cores burning up at 2.53GHz don't feel very slow. 😎

Ruari: If I've set Opera to “Open” .bz2, .gz, .tar and .xz files by automatically plugging them into command switches for untar/install…….does that make me……evil?

Mad Scientist (عادل)qlue # Thursday, September 29, 2011 2:29:11 AM

That's the beauty of choice. 😛 Some hard-core Debian users would cringe at my choice of using the non-free Opera on my Debian based Crunchbang operating system. 😳.

Cutting Spoonhellspork # Thursday, September 29, 2011 2:35:31 AM

And I shudder at the thought of slowing down Crunchbang with five+ other programs to replace everything Opera does. 🥴

Mad Scientist (عادل)qlue # Thursday, September 29, 2011 2:48:18 AM

Actually, Opera is far better suited to creating a cloud OS than Google's Chrome browser. rolleyes. Opera has built-in bit torrent support. A very good built-in email client (which I only use for rss feeds) and Opera link. Then there's Opera Unite. You could do about 60% of all basic office computing with Opera and a good network connection. party. Can you imagine what would emerge if Opera Software were to release their source code under a Gnu licence? 😳

Cutting Spoonhellspork # Thursday, September 29, 2011 4:16:52 AM

A big gap will be closed if/when Opera supports filewriter API and henceforth completes the majority of FileIO/Drag-n-Drop support. Really….it does feel that the desktop product is tantalizingly close to a one-stop solution. Google claims to have already achieved this but ChromeOS is embarrassingly deficient in several key areas when denied network access. Meanwhile Dragonfly demonstrates how amazing a hybrid online/offline application could really be.

And now Opera has the “TV Store” or whatever for their new SDK. This implies hybrid applications, more powerful widgets or extensions, possibly some form of DRM support. With a few added features from the Desktop product, Opera TV could be equal or superior to ChromeOS in many different ways.

Ruarí Ødegaardruario # Thursday, September 29, 2011 6:41:16 AM

He he … guys, I am pretty lax on my blog and really don't mind if people go a little of topic but we have strayed quite far from packaging and dependency management. 😛

Ok, to bring it back on topic, here is a trick that a user with a full Slackware install can use to see what his or her largest (install-size) packages are. It'll help get some ideas about what to consider removing if you do need to conserve a little space.

Issue the following to produce a sorted list of all packages at least 1Mb in size: $ grep "^U.*M$" /var/log/packages/* | sed -r "s|.+/(.+):.+: +(.+)|\2\t\1|" | sort -n

If you are not a regex fan, here is a different way that is a bit easier to read: $ (cd /var/log/packages/ ; grep -x "U.*M" * | awk -F: '{print $3 "\t" $1}' | sort -n)

Ruarí Ødegaardruario # Thursday, September 29, 2011 8:17:37 AM

Originally posted by ruario:

For example, prior to the change of default Slackware package compression format (from gzip to xz) it was not uncommon for some Slackers to hack in support for alternative compression formats themselves.

Just an extra point on this. I still do this myself, not to add stronger compression but instead faster compression. My own installpkg and upgradepkg are hacked to support the extremely quick lzo compression method because I often convert (using an adapted op2slk) various internal builds of Opera Next to this tweaked Slackware format for my own use. I'm more concerned with the speed of conversion, installation and upgrade than the size of the packages themselves. In fact, I usually delete the actual packages immediately after installation as I know I can just recreate them later if needed.

Ian Murrayianmurray # Friday, September 30, 2011 2:12:02 AM

It has never put me off.In fact I love it for that.If ./configure finds a dependency that really need to be there OK I will get it.

Unregistered user # Tuesday, January 14, 2014 2:51:16 AM

Geckgo writes: Thanks for this article. I've had slack running in a VM for a couple weeks now. I'm making an automated install so that when I switch to slack full time on my laptop, I can just run a couple script files and be done with the whole thing, completely packaged and configured the way I want, right down to how new user /home spaces are set up, like my own micro-distro. And if I thrash my system I can have another identical one up and running less than an hour. (sometimes I get lazy when compiling installing things like games and miss something) Just wanted to say, that while far from minimal, /a /d /l /n and /x will give you a nice base system to work with. The only other "dependencies" I've needed so far are select picked from /ap /xap and ../extras and they are part of my scripts :) Doing it this way is teaching me a ton about slackware and linux in general. Yesterday was the first time that I realized colorful ls commands are a result of a bash alias and not the terminal editor :P Learning something new everyday 🙂

Above is the archived comments section. Please leave that unchanged. New comments are welcome, but place them *below* this note. Thank you.

From:
https://docs.slackware.com/ - **SlackDocs**

Permanent link:
https://docs.slackware.com/talk:slackware:package_and_dependency_management_shouldn_t_put_you_off_slackware

Last update: **2014/03/03 14:31 (UTC)**