How to install ANYTHING in Ubuntu!

A graphical guide for all new users with a Windows background using Ubuntu ^{Revision 4}

Having problems installing something The Applications menu on your new Ubuntu operating

system? "Where's the EXE?", "Where do I need to extract this to?", "How do I run it?", "Where did it go?" - have you been thinking questions like these? Don't worry, installing software, themes and other things on Ubuntu is actually very easy! This guide will help you understand with screenshots, instructional videos and to-the-point language.

Translations: Português (Portuguese), Magyar (Hungarian) Coming Soon!

- The package manager
- Installing software with Synaptic
 - The 3 steps: Search, mark and apply
 - But what if my program isn't available through Synaptic?
 - I installed it, but where did my program go?
 - How do I uninstall the program?
- Installing software with the terminal
- Installing a package manually (.deb, .rpm, .tar.gz, .package, klik:// → .cmg, .sh, .bin, .exe, ...)
- Appendix
 - Enabling extra repositories
 - Using CDs as offline repositories
 - Navigating the terminal
 - Adding a launcher/shortcut to your desktop
 - Changing permissions
 - Why I created this guide

Note: The guide was written for Ubuntu 6.06 - The Dapper

Drake using the tools of the Gnome desktop environment as its base. The program names and paths will undoubtedly vary if you're using Kubuntu or Xubuntu.

The package manager

Linux applications are almost all open source ^[1] and they're, unlike typical Windows programs, highly dependant on external libraries to work. You don't have to understand what libraries are, but just that Windows programs typically include parts of libraries in their installers, taking up lots of space after they've been installed because the same libraries have duplicates many places on your harddisk; Linux programs usually don't do this.

Most Linux operating systems have evolved a system where you can download the program, along with any needed dependencies, without having duplicates scattered all over your harddisk saving you lots of space. At the same time, this system allows you to have a central location from which to install and update packages. This system is called the package manager and on Ubuntu you'll meet it in the form of *apt-get, aptitude, Add/Remove..., Update Manager* and *Synaptic.* All these programs are frontends to the same package manager^[2] built right into Ubuntu.

Installing software with Synaptic

Synaptic is a graphical program for installing packages and probably the one you'll feel the most comfortable with. You can launch it from **System** \rightarrow **Administration** \rightarrow **Synaptic Package Manager** (paths will vary depending on your locale, the System menu is the third menu on the menubar at the very top of your screen); as a safety precaution it will ask you for your password before proceeding! It's not because it's dangerous, Ubuntu is just very strict with trying to keep you, and more importantly, non-administrator users, from messing up your system.

Reload Mark All Upgrades	Apply	Properties Search			
	s	Package	Installed Version	Latest Version	Description
Amateur Radio (universe)		2vcard		0.5-lubuntul	A little peri script to convert an adressbook to
sase System		3dchess		0.8.1-11.1	3D chess for X11
ase System (restricted)		3ddesktop		0.2.9-5.1ubuntu1	"Three-dimensional" desktop switcher
ase System (universe)		44bsd-rdist		20001111-6	4.4BSD rdist.
ommunication		6tunnel		0.11rc1-1	TCP proxy for non-IPv6 applications
ommunication (multiverse)		915resolution		0.5-1ubuntu4	resolution modify tool for Intel graphic chipset
ommunication (universe)		9menu		1.8-1.1ubuntu1	Creates X menus from the shell
ross Platform		9wm		1.2-7	emulation of the Plan 9 window manager 8-1/2
ross Platform (multiverse)		a2ps		1:4.13b.dfsg.1-0.1	GNU a2ps - 'Anything to PostScript' converter
ross Platform (universe)		a2ps-perl-ja		1.45-4	perl version of Miguel Santana's a2ps (support
evelopment		abakus		0.90-6ubuntu1	calculator for KDE
evelopment (multiverse)		abc2ps		1.3.3-3	Translates ABC music description files to Posts
evelopment (restricted)		abcde		2.3.99.2-1	A Better CD Encoder
evelopment (universe)		abcm2ps		4.11.8-1	Translates ABC music description files to Posts
ocumentation		abcmidi		20051010-1	converter from ABC to MIDI format and back
ocumentation (multiverse)		abcmidi-yaps		20051010-1	yet another ABC to PostScript converter
ocumentation (universe)		abicheck		1.2-2	binary compatibility checking tool
ditors		abiword	2.4.2-0ubuntu5	2.4.2-0ubuntu5	WYSIWYG word processor based on GTK2
ditors (multiverse)		abiword-common	2.4.2-0ubuntu5	2.4.2-0ubuntu5	WYSIWYG word processor based on GTK2
ditors (universe)	4				
lectronics (multiverse) lectronics (universe) mail (multiverse) mail (universe) Sections Status	No pi	ackage is selected.			

The Synaptic Package Manager

The 3 steps: Search, mark and apply

1. The Search icon in Synaptic First you search for the package you want to install.

Note that there are thousands of themes, applications, libaries and documentation available right away in Synaptic. All of these packages are located on the Ubuntu servers for you to download and update; the package manager essentially works like a kind of improved Windows update that will not only keep your operating system updated, but also all of the non-critical programs you've installed through it. You can find packages by looking for them inside the categories on the sidebar to the left or search for them. Click the *search icon* in the toolbar to search.

The search popup in Synaptic

The search popup that will pop up when you hit search

2. When you've found the package you want to install, right-click on it and mark it to be installed. Most likely it will inform you of a bunch of dependencies that will also be installed in the same procedure; this is all being taken care of automatically! Note that you can also remove packages in the same way (right-clicking and selecting remove instead). Also note that you can mark more than one package to be installed, speeding up the install procedure significantly.



Selecting a package for installation in Synaptic

3. The Apply icon in Synaptic Once you have marked all the packages you want to install,

you can click the *apply icon*. This will download, install and set up everything! It's *that* easy.

Tango video icon See a screencast of the 3-step install process

But what if my program isn't available through Synaptic?

Trust me, it probably is. If it isn't, here are some of the reasons why it's not and how to fix it:

- The Ubuntu package manager gets its package lists from the main Ubuntu repositories, but there are more repositories than just these default ones. There are even more official Ubuntu repositories! Try <u>enabling extra</u> <u>repositories</u> before you give up all hope!
- If you're not connected to the Internet, you're not completely out of luck. There is a chance that your package is <u>available on a CD</u>.
- Even if the package is not available in any repositories, you can still do the old Windows trick: installing it manually. But remember, there are more types of package-formats in Linux than you can think of. You might want to look at the explanation of how you go

about installing a package manually.

I installed it, but where did my program go?

Usually your Applications menu is updated with a launcher to your new program, but sometimes this doesn't happen automatically. Here are some ways to find a link to your new program:

- Install the Debian menu. The Debian The Debian menu menu has a much more thorough list of your installed applications, and it will be available as a category in your existing Applications menu. You need to install the package called *menu-xdg* and possibly restart X (ctrl + alt + backspace) for it to show up.
- They will very likely be available as a terminal command with the same name as the package. Try running the package name as a command in the terminal. Say I've installed the package *muine*^[3] through Synaptic, then I open up the terminal and enter muine ending with a press on the return/enter key. Muine fires up. Note that the application will close when you close the terminal window! To avoid this behavior, press Alt + F2 and the *Run Application* window will show up; Type in muine to start it. Sometimes the command isn't called exactly the same as the package; try typing the beginning letters and then press tab twice. This will either give you the name of the command or a list of names to choose from.
- Right-click on your package in Synaptic, select *Properties* from the menu and click on the tab labeled *Installed Files*. There will be a list of installed programs; the ones installed inside the folder /usr/bin are most likely the name(s) of the command(s).



The installed files of the music player Muine

How do I uninstall the program?

When you want to remove a program, you do exactly the same as when installing - just select *Mark for Removal* instead of *Mark for Installation* in step 2. If you want to remove configuration files as well (maybe you want some weird modifications undone) select *Mark for Complete Removal*. Remember to apply the changes!

Installing software with the terminal

Very often, you'll see other Ubuntu users saying "You can install program ABC with this code ..." and then they'll provide you with a command you can input in the terminal. This not unlike what Synaptic does. In fact, Synaptic uses these commands below the friendly user interface! You can find the terminal at **Applications** \rightarrow **Accessories** \rightarrow **Terminal**. The two commands that you can use are:

sudo apt-get install ABC and sudo aptitude install ABC

ABC is just a fictious package in this case, not a real one. The sudo part of the command means you temporarily grant super-user/administrator rights to the command, provided

you supply a correct user password. It's the same thing that happens when you open up Synaptic, only in the terminal instead! If you run aptitude by itself like this sudo aptitude , you get something that looks like a command-line version of Synaptic.



The Aptitude user interface

It's also possible to search from the command-line like it is in Synaptic. Try this:

apt-cache search ABC or aptitude search ABC

To uninstall a package:

sudo apt-get remove ABC and sudo aptitude remove ABC

Removing configuration files as well:

sudo apt-get remove --purge ABC and sudo aptitude purge ABC

Though the command-line can be scary for new users, as you can see it's fairly simple and straight-forward to use and has many of the same features as Synaptic when it comes to installing software. Some users prefer installing software through the terminal, others don't. You decide for yourself what you like best. Tango video icon See a screencast of the terminal search & install process

Installing a package manually

Are you absolutely sure you can't find the package in <u>Synaptic</u>? Did you try <u>enabling extra repositories</u>? If you've tried all this with little or no success, here's how you do it the Windows-style way. Download a package (<u>.deb</u>, <u>.rpm</u>, <u>.tar.gz</u>, <u>.package</u>, <u>klik:// \rightarrow .cmg, <u>.sh</u>, <u>.bin</u>, <u>.exe</u>) and let's have a look.</u>

Installing a ...

Debian Package (.deb)

When you download a program with the package manager, you actually download Debian packages! It's possible to install individual Debian packages you've downloaded yourself, but unless they're built specifically for Ubuntu, they're not guaranteed to work. Installing them is rather simple in Ubuntu 6.06 Dapper Drake: double-click the package in Nautilus or on your desktop and a package installer will show up:

> Gdebi Debian package installer The Debian Package installer (GDebl)

You simply press *Install Package* to install. If you have a missing dependency, it will inform you of that. It will also inform you if there's a newer version available from the repositories!

Another way to install a Debian package is to use the command dpkg which is what the package manager uses to manipulate Debian packages (or short: debs). The syntax is as follows: if your package is located on your desktop and your username is *carl*, then you install the package *test.deb* with dpkg -i /home/carl/Desktop /test.deb . You need to take care of dependencies yourself, so it's not the optimal way of installing software.

RPM Package (.rpm)

RPM is another popular way of packaging software, and it's used by popular distributions such as Fedora Core, SUSE Linux and Mandriva. RPM is not used by the Ubuntu Package Manager, but there *does* exist a command for converting an RPM into a Deb; this doesn't mean that any RPM will work on your system, though! The same program can also install the RPM directly so that you won't have to do this yourself. The command is not available right away so you'll need to install it yourself the package is called *alien* and is of course available through Synaptic. If the user carl wants to install an RPM called *test.rpm* located on his desktop, he will enter sudo alien -i /home/carl/Desktop/test.rpm .

Desktop Theme (.tar, .tar.gz, .tgz, .tar.bz, ...)

Installing themes^[4] is relatively painless in Gnome. You open the Theme Preferences which you'll find at **System** \rightarrow **Preferences** \rightarrow **Theme**. With this application you can change icons, controls and window borders to your liking. To install your theme, simply drag and drop the package onto the Theme Preferences window and confirm the dialog window that pops up. To use your new theme, edit one of the existing themes to use your new icons, controls or window borders.

Installing a new desktop theme Click Install to install the new desktop theme

Login Screen Theme (.tar, .tar.gz, .tgz, .tar.bz, ...)

Installing themes for your login screen is as simple as installing desktop themes. You open up Login Window Preferences in **System** \rightarrow **Administration** \rightarrow **Login Screen** and drag and drop your theme onto the window. Confirm the dialog window that pops up. To use your new theme, select it in the list of themes.

Installing a new login screen theme Click Install to install the new login screen theme

Source Package (.tar, .tar.gz, .tgz, .tar.bz, ...)

Note: not all files with an extension named .tar, .tar.gz, and so on are archives with source code in them - they might be precompiled! If the archive is precompiled, there should be an installer or a binary executable inside it.

Sometimes all you've got is a package full of uncompiled source code. Luckily, you don't need to be a programmer to know how to compile and install a package with source code. Back in the old days, this was the only way to install software on Linux and there *is* a standard way of installing these files. It will not work in every case, but it will in most (if you have the right dependencies installed). To compile a package you must first extract it somewhere. This is easily done, simply right-click on the package and select *Extract Here*.

Extracting a package

Tango video icon See a screencast of package contents being extracted

To proceed you must have the compiler tools installed. They all come with the package *build-essential*, available in Synaptic. When you're sure you have the compiler tools installed, you fire up the terminal and change directory to the one you've just extracted (if you're not sure how to do that see: *Navigating the terminal*.

When you're in the correct directory you execute a configure script: ./configure . The purpose of the configure script is usually to check for dependencies and then create the makefile. If the script fails for some reason and tells you to install certain packages, look up the names in Synaptic (**Important!** If you find packages in Synaptic named almost the same but with a *-dev* extension, remember to install those as well! They're development packages and are needed for compiling). Don't worry if it complains that there is no configure script

- many packages don't come with one! Then you compile it with make and after it's been compiled you can install it. There are two ways:

Normal install: If you want to install it the normal "primitive" way, type sudo make install . To remove the temporary files you run make clean . To uninstall the program you run sudo make uninstall . These two clean-up commands don't always work, though, the programmer needs to have enabled them.

Package manager install: If you want to install it in a way that means it can be easily removed from inside the package manager, first install the package *checkinstall*. To install the package type sudo checkinstall . This will take slightly longer than a normal install and quite possibly you'll have to supply a description of the application yourself (and edit the other information slightly). If the need arises, this will be easy to take care of from inside the checkinstall program.

Tango video icon See a screencast of a package being installed from source

Autopackage (.package)

The Autopackage format is supposed to be a "Linux distribution"-neutral way of installing packages. It uses its own self-contained package management tool, which is download from the Internet and installed along with the first Autopackage you install. To install an autopackage called test.package located on user Carl's desktop, first make sure it has <u>permission to execute</u> in your file-system, then run <u>/home/carl/Desktop/test.package</u>. The installer will ask you some questions, likely what your password is.

Klik package (klik:// → .cmg)

Klik is an online software repository that uses its own

protocol called klik:// (similar to http://) to allow you to install a package from their site by clicking on a hyperlink in your web browser. To use Klik you must first install the packages *binutils libstdc++5 rpm gnome-about* using Synaptic and then run this command in the terminal: wget klik.atekon.de/client/install -O -|sh which downloads and installs the Klik client. Klik completely bypasses both your package manager and your file-system; everything you need to run the program is included in a .cmg-file located on your desktop after the installation is done. You simply double-click the file to start your application. To remove it again, simply delete the .cmg-file

Shell Script Installer (.sh, .bash, ...)

You can run the shell script inside a terminal with the command sh . If the script is called *test.sh* and is on the desktop of user carl, you can install it with sh /home/carl /Desktop/test.sh . Keep in mind that the script might not have permission to execute in your file-system.

Custom Binary Installer (.bin, ...)

If the installer is called *test.bin* and is located on the user carl's desktop, you can run it inside your terminal with /home/carl/Desktop/test.bin . Keep in mind that the installer might not have <u>permission to execute</u> in your file-system.

Windows Executable (.exe)

If you, after having searched around the Internet for a Linux version or a viable Linux replacement for the Windows program you want to install, find that there is no Linux program that will replace it, there is a slight chance the Windows executable will run on Linux^[5]. This is not a proper solution to your problem, not in any way, but for some people it's the only way. To run Windows executables you need to install a package called *wine*. When that is done, run the command wine PATH in the terminal where PATH is the path to your EXE. If the user carl has an EXE called test.exe inside his home folder, he'll run the command wine /home/carl/test.exe to execute it. Be adviced that running Windows programs in WINE is often very buggy and probably won't work to your satisfaction; very often it doesn't work at all!

If the executable you ran was an installer wizard, your program will be installed in a hidden folder located inside your home folder. If the user carl has installed a program called *Test*, it will probably be installed to the folder **"/home/carl/.wine/drive_c/Program Files/Test"** (remember to include quotes around paths with spaces in them when typing them in a terminal). EXEs from inside this folder can be run with *wine*. You might want to <u>create</u> <u>at launcher/shortcut</u> for your desktop to easily start up your app. Here is a Windows program run with *wine*:

> The Windows application Graph The Windows application Graph

Appendix

The following isn't directly connected to installing software and themes.

Enabling extra repositories

On a standard Ubuntu installation, Ubuntu is configured to use the *main* repository. There are however, other official repositories (or sections on the Ubuntu server) that aren't available right away. There is one called *Universe* which is the largest one. It's a pool of community-maintained software, but it is not officially supported by Ubuntu. There is also a section called *Multiverse* which has software under questionable licences. The third section is called *Restricted* and is a very small pool of software with restricted copyright.

To enable the rest of the Ubuntu repositories you open Synaptic and select an option in its menubar: **Settings** \rightarrow **Repositories**. Here is a list of the current repositories. To enable the missing section select each of the packages that are labeled *binary*, click on *Edit* and tick the boxes outside the sections of the Ubuntu repositories you want to enable.

Enabling the universe and multiverse repositories Enabling the universe, restricted and multiverse repositories

Tango video icon See a screencast showing how to enable the extra repositories

When you're done, Synaptic will probably ask you to reload your list of packages; agree to do that. Now your list of available packages should have increased significantly.

Using CDs as offline package repositories

The best way to install new software in Ubuntu is to be connected to the Internet, but sometimes this is not possible. When you install Ubuntu the first time, your install CD should have been added as a repository. If it isn't, you can add it from the same window you <u>enable extra repositories</u> from. There's a button labeled *Add CDrom*; press this, insert your install CD and it will be added to the repositories.

> Adding a CD to the package manager Adding a CD to the package manager

You can now install software through Synaptic without being connected to the Internet, provided the install CD is inserted. Note that the install CD has software solely from the main repository, not Universe, Multiverse or Restricted! You now can download the <u>entire Universe and Multiverse repositories</u> on a set of DVD's and CD's courtesy of a volunteer effort. There is an ongoing project to create an Addon CD or DVD with select packages from the other sections of the Ubuntu repositories. You can <u>download a preliminary CD ISO file</u> for Ubuntu 5.10 'Breezy Badger' and try it out if you want to, but this guide will not go further into the subject^[6].

Navigating the terminal

The standard terminal on Ubuntu is Gnome Terminal which can be found in **Applications** \rightarrow **Accessories** \rightarrow **Terminal**. A terminal is in a way very similar to a file manager in that it's always inside a specific folder and is able to navigate to other folders and do regular file management. By default it will be inside your *home* folder when you run it. To confirm that your terminal is indeed browsing your home folder, type pwd ending with a press on enter. The *pwd* command will output the path to the current folder.

To see a list of files and directories inside the current directory, run the command Is . If you want to navigate up the directory tree run cd ... If you want to navigate down the directory tree run cd NAME where *NAME* is the name of the folder you want to navigate to. Example: if Tom is inside his home folder and there's a directory called *test* inside it, he will run cd test to change directory. If he wants to go back he can run cd ... I he ever gets lost he can run cd by itself; this will take him back to his home folder.

Adding a launcher/shortcut to your desktop

These are well-known from Windows. Launchers are shortcuts to your application allowing you to easily run it. To add a launcher, right-click somewhere on your desktop and select *Create Launcher....* This will open a dialog from which you can enter information about the launcher. Remember to enter a name as well as a path to the executable. This is what carl would enter if he wanted a launcher for the executable named *test* located in his home folder:

> Creating a new launcher Creating a new launcher

Tango video icon See a screencast showing how to create a launcher

Changing permissions

To change permissions so that the file is executable, you right-click on the file on the desktop and select *Properties*; a

window will open. Click on the tab labeled *Permissions* inside the window. There will be some boxes you can tick which tell the system what you're allowed to do with the file. Tick the box that says *Execute* outside the label that says *Owner*.

> Changing permissions on a file Permissions are tied to every file in the file-system

You can also run the command chmod +x /home/carl /Desktop/test.bin to make the file executable.

Tango video icon See a screencast showing a file being made executable

Why I created this guide

The easiest way to help a new user install something is to provide a command that will accomplish it. That is *not* in my opinion the best way to make a new user understand how Ubuntu works. It's also sad how you can leave the impression that Ubuntu *needs* to be used through a command-line for the simplest of tasks, when capable graphical tools exist for the same purposes. I hope this guide will help new users understand everything a little bit better. As they say in China: "Give a man a fish and you feed him for a day. Teach a man to fish and you feed him for a lifetime."

Changelog

- Revision 1 (2006-05-06): Changed the introduction slightly, added list of contents, added part about checkinstall.
- Revision 2 (2006-05-14): Linked to all important sections at the top of the page (BTW thanks to the bloke who <u>put</u> <u>this on digg.com</u> and to all those who dugg it!).
- Revision 3 (2006-05-19): New URL, new introduction (+ disclaimer about the Gnome-centrism), experimenting with instructional screencasts in the GIF format (recorded with <u>Byzanz</u>).
- Revision 4 (2006-07-28): Explained about precompiled archives and development packages in the source

package section, moved the explanation of permissions to its own section in the appendix, added link to the DVD repository project, added link to Portuguese translation, added short description of the Autopackage format and added short description for installation of Klik packages.

Created by Simon Gray © 2006

Notes:

- 1. Open source implies that the software is available under an open source license, meaning anyone is free to look at, modify, copy and redistribute the code under the terms of the license. Open source is considered the most transparent and effective way of developing software today. Ubuntu is entirely made up of open source software (abbreviated OSS).
- 2. Advanced Packaging Tool or APT is what empowers many popular Linux distributions. It was first seen in Debian (that's where the name of the packaging format, deb, came from) but is now also used by other popular distributions such as Ubuntu, Knoppix, Linspire and Mepis. Another popular package manager is RPM (RPM Package manager, a recursive acronym) which is used by Fedora Core, SUSE Linux and Mandriva.
- 3. Muine is a music player and a great way to enjoy your music. It has a simple interface, is very intuitive, focuses on listening to albums and doesn't come with a lot of statistical information. It's light as a feather and makes you enjoy music, without the music managing gaining all your focus. If you want a killer music jukebox with more features than you can imagine, take a look at Amarok instead.
- If you're looking for themes, take a look at <u>Gnome Look</u> or <u>art.gnome.org</u>. The window border theme I used for the screenshots in this guide is called *Shiny Cairo* and can be obtained at the <u>Ubuntu Forums</u>.
- 5. The WINE developers keep a very handy database where the users can submit small reviews of their experiences with all sorts of programs and games when run with WINE. It also includes tips on how to get some of the more difficult applications running in WINE. If you're a web developer and need to see what your sites look like in Internet Explorer 6, there *is* a script available for <u>installing it</u>.
- As of now, the addon CD is entirely unofficial and created by a some very helpful users in the Ubuntu community. There is work on creating a <u>"Breezy Badger" version</u>, but no "Dapper Drake" is being developed on yet! <u>See the progress so far</u>.