Installing Software in Ubuntu

Introduction

Installing software in Ubuntu is easy, and we will show you how to do it.

By default, many useful programs are already installed when you put Ubuntu onto your computer. However, you may need a particular piece of software that serves a purpose not served by the default applications. You might just want to try an alternative program to one which is already installed. In other words, you need new software.

What is a package?

Software is a very broad term, and is generally taken to mean a program which you can run on your computer. However, such programs often need other resources to work. When you install software, thousands of files may be required just to let the program start! When you think that they all have to be put in exactly the right location, and some of those files may need to be changed depending on what type of computer you have, it can all get very complicated. Luckily, Ubuntu can look after this complexity.

Ubuntu uses packages to store everything that a particular program needs to run. A 'package', then, is essentially a collection of files bundled into a single file, which can be handled much more easily. In addition to the files required for the program to run, there will be special files called installation scripts, which copy the files to where they are needed (amongst other things).

Source packages are simply packages which just include source code, and can generally be used on any type of machine if the code is compiled in the right way. Binary packages are ones which have been made specifically for one type of computer, or architecture.

Ubuntu supports the x86 (i386 or i686), AMD64 and PPC architectures. The correct binary packages will be used automatically, so you don't have to worry about picking the right ones. To find out which one you are using, open the Terminal, type ***uname -m*** then hit the enter key.

Package Dependencies

Programs often use some of the same files as each other. Rather than putting these files into each package, a separate package can be installed to provide them for all of the programs that need them. So, to install a program which needs one of these files, the package containing those files must also be installed. When a package depends on another in this way, it is known as a package dependency. By specifying dependencies, packages can be made smaller and simpler, and duplicates of files and programs are mostly removed.

When you install a program, its dependencies must be installed at the same time. Usually, most of the required dependencies will already be installed, but a few extras may be needed, too. So, when you install a package, don't be surprised if several other packages are installed too - these are just dependencies which are needed for your chosen package to function properly.

Package Managers

A package manager is an application which handles the downloading and installation of packages. Ubuntu includes a few package managers by default, and which one you use depends on how advanced the package management tasks are that you want to achieve. Most people will only need to use the most basic package manager, the Add/Remove tool, which is very easy to use.

Software Channels/Repositories

Where can you get packages from?

Ubuntu stores all of its packages in locations called software channels or repositories. A software channel is simply a location which holds packages of similar types, which can be downloaded and installed using a package manager. A package manager will store an index of all of the packages available from a software channel. Sometimes it will 're-build' this index to make sure that it is up to date and knows which packages have been upgraded or added into the channel since it last checked.

There are four Ubuntu software channels for each architecture - Main, Restricted, Universe and Multiverse. Each has a different purpose. By default, only packages from Main and Restricted can be installed.

In addition to the official Ubuntu repositories, it is possible to use third party repositories. Be careful, though - some are not compatible with Ubuntu and using them may cause programs to stop working or may even cause serious damage to your installation.

Installing A Package

1- From the Software Center (12.10 Ubuntu version) or (Synaptic Package Manager in 10.10 and later versions).

2- From the Web browser if enable clicking on a hyper-link to **apt:applicationname AptURL** will allow ubuntu to install application.

3- From the shell (BASH) : apt-get install/remove/purge applicationname

4- If you downloaded a Debian Package (applicationname.**deb**) and wanted to install it (First install sudo apt-get install gdebi.) then use the file browser and double click on the package to install it.

5- Using **dpkg** to install packages: dpkg is a command-line tool used to install packages. To install a package with dpkg, open a Terminal and type the following:

```
cd directory
sudo dpkg -i package_name.deb
```

Note: replace directory with the directory in which the package is stored and package_name with the filename of the package.

It is recommended that you read the dpkg manual page before using dpkg, as improper use may break the package management database. To view the manual page for dpkg, open a Terminal and type man dpkg.

Glossary

apt: The 'Advanced Package Tool', the program on which Ubuntu's Package Managers are based. apt handles the more complicated parts of package management, such as maintaining a database of packages.

Architecture: The type of processor the computer uses is referred to as its architecture.

Binary Package: A package which contains a program suitable for one particular architecture.

deb: A .deb file is a Ubuntu (or Debian) package, which contains all of the files which the package will install.

Dependency: A dependency is a package which must be installed for another package to work properly.

Package Manager: A program which handles packages, allowing you to search, install and remove them. E.g. Add/Remove...

Repository/Software Channel: A location from which packages of a similar type are available to download and install.

Source Package: A package which contains the original code for a program, which must be compiled to be usable on a particular architecture.

References

https://help.ubuntu.com/community/InstallingSoftware