

# Ubuntu 18.04 LTS for Human Beans

This page collects information to have Ubuntu 18.04 LTS (Xenial Xerus) in a laptop or desktop as usable as possible for Human Beans willing to use it for production.

## 1.1. Enable Ubuntu Partners

First enable "partners" repos, since we will use it later to install skype, at least.

### Contents of the updated `/etc/apt/sources.list`



```
## Uncomment the following two lines to add software from Canonical's
## 'partner' repository.
## This software is not part of Ubuntu, but is offered by Canonical and the
## respective vendors as a service to Ubuntu users.
deb http://archive.canonical.com/ubuntu bionic partner
deb-src http://archive.canonical.com/ubuntu bionic partner
```

## 1.2. Extra repositories

Useful extra repositories

### Command on a console



```
xavi@penguinbookpro:~$ sudo su
add-apt-repository -y ppa:nilarimogard/webupd8
add-apt-repository -y ppa:danielrichter2007/grub-customizer
add-apt-repository -y ppa:webupd8team/java
add-apt-repository -y ppa:bit-team/stable
add-apt-repository -y ppa:jtaylor/keepass
add-apt-repository -y 'deb https://cloud.r-project.org/bin/linux/ubuntu bionic-
cran35/' # main binary packages for R 3.5.x
add-apt-repository -y ppa:marutter/c2d4u3.5 # extra binary packages for R 3.5.x
from the usual marruter repo
add-apt-repository -y ppa:ataarea/telegram
add-apt-repository -y ppa:maarten-baert/simplescreenrecorder
add-apt-repository -y ppa:yg-jensge/shotwell
add-apt-repository -y ppa:mkusb/ppa
add-apt-repository -y ppa:hamishmb/myppa
add-apt-repository -y ppa:utappia/stable
add-apt-repository -y ppa:gezakovacs/ppa # for unetbootin
add-apt-repository -y ppa:kubuntu-ppa/backports # upgrades to major versions of kde
apps
```

```
# Add the release PGP keys for syncthing repo:
apt install curl
curl -s https://syncthing.net/release-key.txt | apt-key add -
# Add the "release" channel to your APT sources for updated-twice-per-month
syncthing daemon:
echo "deb https://apt.syncthing.net/ syncthing release" | tee
/etc/apt/sources.list.d/syncthing.list
# Next 3 lines are for syncthing-gtk
sh -c "echo 'deb
http://download.opensuse.org/repositories/home:/kozec/xUbuntu_18.04/ /' >
/etc/apt/sources.list.d/home:kozec.list"
wget -nv
https://download.opensuse.org/repositories/home:kozec/xUbuntu_18.04/Release.key -O
Release.key
apt-key add - < Release.key

# Add the key for the new repo for R 3.5.x from cloud.r-project.org
apt-key adv --keyserver hkp://keyserver.ubuntu.com:80 --recv-keys E084DAB9 #
marutter
apt-key adv --keyserver hkp://keyserver.ubuntu.com:80 --recv 089EBE08314DF160 #
ubuntugis-stable

# Google Earth repo
sh -c 'echo "deb [arch=amd64] http://dl.google.com/linux/earth/deb/ stable main" >>
/etc/apt/sources.list.d/google.list'
# Setup the Google Key with:
wget -q -O - https://dl-ssl.google.com/linux/linux_signing_key.pub | sudo apt-key
add -

# Update packages list again, just in case
apt-get update
```

Press enter each time when requested.

After keys are imported, you can quit from the root session with "exit":



```
root@computer:/home/xavi# exit
exit
xavi@computer:/home/xavi#
```

## 1.3. Install with interaction

Install programs which require interaction, like a confirmation step, a new password or user interaction of some sort.

- mysql-server, for the webserver in localhost, etc.
- phpmyadmin, to manage mysql databases easily through a web browser
- apachetop, for monitoring apache in real time
- ttf-mscorefonts-installer, required by other packages (PlayOnLinux, maybe?)
- a few others like java, and network filesystem network packages

#### Command on a console



```
sudo apt-get install -y ttf-mscorefonts-installer nfs-common libnss-winbind winbind
cifs-utils oracle-java8-installer libdvd-pkg
```

If you need a webserver & php/mysql:

#### Command on a console



```
sudo apt-get install -y mysql-server phpmyadmin apachetop postfix taskel
sudo taskel install lamp-server
sudo mysql_secure_installation
```

if you don't run the `mysql_secure_installation` command, mysql root will not have password set yet, and you won't be able to access mysql through phpmyadmin yet.

But for some reason, Ubuntu 18.04 comes with a different way to authenticate the root mysql user, and you may end up with problems to login to mysql with the password you just indicated in the steps above. Therefore, if that is your case also, you may need to run these steps to setup mysql root password as usual in earlier Ubuntu LTS versions:

<https://linuxconfig.org/how-to-reset-root-mysql-password-on-ubuntu-18-04-bionic-beaver-linux><sup>[1]</sup>

## 1.4. Install the rest in program sets

You can then install all the rest of the required and useful software for common tasks (including video editing, screencasting, etc). Note that the first package names until wcatalan are localization packages for our local language, and you can skip them or adapt for your language, etc.

#### Command on a console



```
sudo apt-get install -y apt-show-versions backintime-common backintime-gnome
bluefish bluefish-plugins bzip-explorer cairo-dock cairo-dock-plug-ins cheese
chromium-browser classicmenu-indicator corkscrew create-resources printer-driver-
cups-pdf diodon filezilla firefox-locale-ca flashplugin-installer flvstreamer gdebi
geany geany-plugins gimp-plugin-registry glib1.2-gst-plugins-bad-1.0 git git glabels
```

```
gnome-alsamixer gnupg2 gparted grub2-splashimages grub-customizer gscan2pdf
gstreamer1.0-plugins-good gstreamer1.0-fluendo-mp3 gstreamer1.0-plugins-bad
gstreamer1.0-plugins-ugly gtk-recordmydesktop gufw guvcview gwenview handbrake
hibernate htop hyphen-ca icatalan icc-profiles-free indicator-cpufreq inkscape
ispell kazam kde-l10n-ca kdenlive keepass2 keepass2-doc kile kipi-plugins krename
krusader kupfer language-pack-ca language-pack-ca-base language-pack-gnome-ca
language-pack-gnome-ca-base language-pack-kde-ca lftp libcurl4-gnutls-dev libdbd-
mysql libgtk2.0-dev libmysqlclient-dev libreoffice-help-ca libreoffice-l10n-ca
libsyntax-highlight-engine-kate-perl linux-firmware luckybackup mailutils mc meld
mobile-atlas-creator mythes-ca nautilus-dropbox nautilus-scripts-manager
oggvideotools ogmrip ogmtools openjdk-8-jre openshot openshot-doc openssh-server
p7zip p7zip-full p7zip-rar pavucontrol pdfmod pdfsam pepperflashplugin-nonfree
pidgin pidgin-plugin-pack pidgin playonlinux powernap qemu-system qshutdown
qtgstreamer-plugins-qt5 scribus scribus-doc scribus-template sendmail shutter
simplescreenrecorder sshfs synaptic syncthing telegram terminator thunderbird
thunderbird-locale-ca totem-plugins totem-plugins-extra transmageddon ubuntu-
restricted-addons ubuntu-restricted-extras unity-tweak-tool unrar uswsusp variety
virt-manager virtualbox virtualbox-dkms vlc vpx-tools wcatalan webp wine-stable
winetricks x2goclient xauth xfonts-100dpi xfonts-75dpi xfonts-base xsane xul-ext-
lightning gedit-plugins kio-extras-data shotwell testdisk geeqie kwalletmanager
firefox libvdpau-va-gll soundconverter keychain mtp-tools mkusb mkusb-nox usb-pack-
efi ddrescue-gui torbrowser-launcher icedtea-8-plugin pdfposter posterazor
libgstreamer-plugins-base1.0-dev btrfs-progs btrfs-tools libfsntfs-utils
ucaressystem-core net-tools python-gpg nautilus-dropbox simple-scan youtube-dl
youtube-dlg syslinux-utils unetbootin libgdal-dev libproj-dev libgeos-dev
libudunits2-dev libv8-dev libcairo2-dev libnetcdf-dev libgeos++-dev semantik dia
alien pigz
```

If you need webserver and other typical programs intended for servers, you can install this other set too:

#### Command on a console



```
sudo apt-get install -y php-sqlite3 postfix rabbitvcs-cli rabbitvcs-core
rabbitvcs-gedit rabbitvcs-nautilus rapidsvn subversion php-memcache php-gettext
php-intl php-mbstring php-pspell php-zip composer php-curl php-xmlrpc php-intl php-
mysql poppler-utils php-memcached memcached
```

If you want to have locale packages such as Catalan, you might be intersted in installaing these extra packages:

#### Command on a console



```
sudo apt-get install -y aspell-ca hunspell-ca
```

And for Spanish:

#### Command on a console



```
sudo apt-get install -y aspell-es hyphen-es ispanish language-pack-gnome-es  
myspell-es mythes-es wspanish
```

## 1.4.1. R

Per omisió, Ubuntu 18.04 ve amb R 3.4.x. Però en els passos previs d'aquestes instruccions, s'han afegit ja els repositoris per a que la instal·lació d'R es faci amb la versió 3.5.x, que porta algunes millores interessants, desenvolupades inicialment a la branca ALTREP<sup>[2]</sup> d'R.

Per poder instal·lar adequadament paquets d'R des de dins de la consola d'R et caldran paquets extra de sistema. I t'aniran bé algunes eines habituals associades (git, svn, etc). Pots posar les dependències habituals amb:

#### Command on a console



```
sudo apt-get install -y r-recommended r-cran-xml libgraphviz-dev libcairo2-dev r-  
cran-cairodevice freeglut3 freeglut3-dev r-cran-rgl r-cran-rgl r-cran-misc3d  
libx11-dev libxt-dev libcurl4-gnutls-dev libxml2-dev r-cran-xml libgraphviz-dev  
libcairo2-dev bwidget tk-table libv8-dev r-cran-rjava libmpfr-dev libc6 libssl-dev  
texlive-latex-extra texlive-lang-spanish libx11-dev libxml2-dev libxml2:i386 libxt-  
dev r-cran-misc3d subversion git texmaker tk-dev unaccent xvfb libgdal-dev libproj-  
dev r-cran-rmysql libmagick++-dev r-cran-rcolorbrewer r-cran-doparallel libssh2-1-  
dev libudunits2-dev libgdal-dev libgeos-dev libproj-dev libjq-dev libprotobuf-dev  
protobuf-compiler libssl-dev libcairo2-dev libglpk-dev dos2unix pandoc r-cran-rmio  
r-cran-gstat r-cran-maps r-cran-mapdata r-cran-ncdf4 r-cran-sp r-cran-raster r-  
cran-geor r-cran-ggmap r-cran-leaflet r-cran-rjags r-cran-snow r-cran-ggplot2 r-  
cran-igraph r-cran-lme4 r-cran-devtools r-cran-roxygen2 r-cran-rjava r-cran-xlsx
```

Una llista de paquets habituals d'R que et pot interessar tenir instal·lats de partida (un cop iniciis una sessió d'R) són:

#### Command on an R console



```
install.packages(c("devtools", "plotly", "Nozzle.R1", "VennDiagram", "stringr",  
"xml2", "data.table", "doParallel", "xtable", "plyr", "dplyr", "reshape2", "rjson",  
"d3heatmap", "htmlwidgets", "googleVis", "doMC", "knitr", "checkpoint",  
"XLConnect", "sem", "rmarkdown", "relimp", "effects", "aplpack", "flexdashboards",
```

```
"addinslist", "tidyverse", "magick", "webshot", "pacman", "officer", "flextable",  
"huxtable", "rio", "sf", "png", "rsvg", "xml2", "sparklyr", "rgdal", "rosm",  
"tmap"))
```

Si estàs actualitzant des d'una versió d'R anterior, et poden ser útils aquestes instruccions de terminal de sistema:

#### Command on a console



```
sudo chmod 777 /usr/lib/R/site-library /usr/lib/R/site-library/* -R  
sudo chmod 777 /usr/lib/R/library /usr/lib/R/library/* -R  
sudo chmod 777 /usr/share/R/doc/html/* -R
```

I aquestes altres de consola d'R:

#### Command on an R console



```
update.packages(ask = F, lib="/usr/lib/R/site-library", checkBuilt = TRUE)
```

Cal tenir present que per omissió RStudio demana la contrasenya de Gitlab/Github cada vegada que vols fer un **push** dels commits locals cap

## 1.4.2. Ús de credencials git via RStudio

Per omissió RStudio demana la contrasenya de Github cada vegada que vols fer un **push** dels commits locals cap a un repositori remot (com Gitlab o altres). Per que no ho demani cada vegada, pots dir-li que empri l'anell de claus del sistema per a recuperar les credencials git, amb les següents instruccions:

#### Instruccions en terminal



```
sudo apt-get install libgnome-keyring-dev  
sudo make --directory=/usr/share/doc/git/contrib/credential/gnome-keyring  
git config --global credential.helper /usr/share/doc/git/contrib/credential/gnome-  
keyring/git-credential-gnome-keyring
```

# 1.5. Boot-repair (in multi-boot environments)

These instructions will save you some headache in a multi-boot environment when one OS changes the boot menu and doesn't allow you to boot in your preferred OS. You can boot from an Ubuntu live CD/DVD/USB, and install the boot-repair program from its own repo, with:



```
sudo add-apt-repository ppa:yannubuntu/boot-repair
sudo apt-get update
sudo apt-get install -y boot-repair && boot-repair
```

More information:

<https://help.ubuntu.com/community/Boot-Repair><sup>[3]</sup>

Non installed yet:

- jitsi -> installed by hand from their website, v 2.8.x
- linux-firmware-nonfree
- gstreamer1.0-plugins-bad-multiverse -> gstreamer1.0-plugins-bad

Other useful programs installed by hand in a later stage:

- rstudio
- chrome
- jitsi
- gdiskdump
- phpstorm
- acroread
- canon scanGear MG3000. See Canon PIXMA MG3000 Printer and Scanner
- "Master pdf editor"  
<http://www.webupd8.org/2014/02/modify-pdf-files-in-linux-with-master.html><sup>[4]</sup>  
(<http://code-industry.net/free-pdf-editor.php><sup>[5]</sup>)
- elastic search

You can test email sending in the command line with:



```
echo testing | mail -s test_subject youremail@example.com
```

## 1.5.1. Main changes noticed for human beans

Good:

- Bluetooth works by default with most devices
- Battery icon displays battery info for connectd bluetooth devices using batteries (mouse, keyboard, etc)
- Icon to edit microphone volume levels shows up automatically on the top bar when the microphone is in use

Confusing:

- "USB startup creator" doesn't let you choose a persistent partition file when creating bootable usb disks with ubuntu distros.

## 1.5.2. Extra R packages to be installed

Usual R packages that are useful in many cases. You can install them in an R console in a terminal (Type "R" in the terminal to launch the R console), or through RStudio command line.

Run in an R console



```
if (!require(devtools)) install.packages("devtools"); library(devtools)
# Check Availability of package manager "PacMan" (and install it, if missing)
# if you have R < 3.5
if (version$major==3 && version$minor < 5) {
  # Instal·la devtools si et cal
  if (!require("devtools")) install.packages("devtools"); require("devtools")
  if (!require("pacman")) {
    cat("R Version: ", paste0(version$major,".",version$minor), ". ")
    cat("Tens una versió de R anterior a 3.5, per tant, instal·lem la versió
corresponent de PacMan (0.4.1)\n")
    # Instal·la la darrera versió de pacman (0.4.1) que anava abans de la que va
només amb R 3.5+ (pacman v0.5)
    install_version("pacman", version = "0.4.1"); require("pacman")
  }
} else {
  # if you have R 3.5 or higher
  if (!require("pacman")) install.packages("pacman"); require("pacman")
}
p_load("devtools", "plotly", "Nozzle.R1", "VennDiagram", "stringr", "xml2",
"parallel", "data.table", "doParallel", "xtable", "plyr", "dplyr", "reshape2",
"rjson", "d3heatmap", "htmlwidgets", "googleVis", "doMC", "knitr", "checkpoint",
"sem", "rmarkdown", "relimp", "aplpack", "addinslist", "tidyverse", "magick",
"webshot", "pacman", "officer", "flextable", "huxtable", "RODBC", "leaflet",
```



```
"gitlabr", "colorspace", "mapdeck", "rsvg", "png", "cartography", "magick",  
"rJava", "rio", "sf", "png", "rsvg", "xml2", "sparklyr", "rgdal", "rosm", "tmap")  
# library(devtools);if(!require(rCharts)) install_github('rCharts', 'ramnathv') #  
deprecated approach of doing fancy charts - kept here for historical purposes only
```

## 1.5.3. Install new php versions and switch between them

Ubuntu 18.04 comes with php 7.2, and some php applications might still fail with php 7.2 .Therefore, in some cases, it might be sensible to have both php 7.0 and php 5.x, so that you can choose which version to use for your needs at any time. Or even better, 7.1. You need to add a new repository to have other versions of php available, and you will require the equivalent branch for a patched apache2 that will work with the other php version repository.

You can do so with:



```
xavi@computer# sudo su  
root@computer# add-apt-repository ppa:ondrej/php  
root@computer# add-apt-repository ppa:ondrej/apache2  
root@computer# apt-get update  
root@computer# apt-get install php5.6 php5.6-mysql php-gettext php5.6-mbstring php-  
xdebug libapache2-mod-php5.6 php5.6-curl php5.6-gd php5.6-mcrypt php5.6-xml  
php5.6-xmlrpc libphp5.6-embed php-memcache php5.6-intl php5.6-zip php5.6-sqlite3  
php5.6-zip  
root@computer# apt-get install php7.1 php7.1-mysql php-gettext php7.1-mbstring php-  
xdebug libapache2-mod-php7.1 php7.1-curl php7.1-gd php7.1-mcrypt php7.1-xml php-  
xml-parser php7.1-xmlrpc libphp7.1-embed php-memcache php7.1-intl php7.1-zip  
php7.1-sqlite3 php7.1-zip  
root@computer# apt-get install php7.2 php7.2-mysql php-gettext php7.2-mbstring php-  
xdebug libapache2-mod-php7.2 php7.2-curl php7.2-gd php7.2-xml php7.2-xmlrpc  
libphp7.2-embed php-memcache php7.2-intl php7.2-zip php7.2-sqlite3 php7.2-zip  
root@computer# apt-get install php7.3 php7.3-mysql php-gettext php7.3-mbstring php-  
xdebug libapache2-mod-php7.3 php7.3-curl php7.3-gd php7.3-xml php7.3-xmlrpc  
libphp7.3-embed php-memcache php7.3-intl php7.3-zip php7.3-sqlite3 php7.3-zip  
root@computer# apt-get install php7.4 php7.4-mysql php-gettext php7.4-mbstring php-  
xdebug libapache2-mod-php7.4 php7.4-curl php7.4-gd php7.4-xml php7.4-xmlrpc  
libphp7.4-embed php-memcache php7.4-intl php7.4-zip php7.4-sqlite3 php7.4-zip
```

Eventually Installing both php5.6 & php7.\* resulted in a lot of complaining from apt and lots of conflicts. The first suggested resolution was to remove all the stock php5 packages so that PHP 5.6 could be installed - so I just accepted the first suggestion.

The config files are all in `/etc/php/5.6` and `/etc/php/7.*` respectively - inside here is where you can configure which extensions are loaded, set the ini settings, and everything else for each version in isolation.

So to switch from php 5.6 to php 7.2 you need to do two things:



```
# For php in web apps
user@computer:/# sudo a2dismod php5.6; sudo a2enmod php7.2; sudo service apache2
restart
# For php-cli in the command line
user@computer:/# sudo ln -sf /usr/bin/php7.2 /etc/alternatives/php
user@computer:/# php -v
# PHP 7.2.x (...)
# Copyright (c) 1997-2016 The PHP Group
# Zend Engine v3.0.0, Copyright (c) 1998-2016 Zend Technologies
```

or from php7.2 to php5.6:



```
# For php in web apps
user@computer:/# sudo a2dismod php7.2 ; sudo a2enmod php5.6 ; sudo service apache2
restart
# For php-cli in the command line
user@computer:/# sudo ln -sf /usr/bin/php5.6 /etc/alternatives/php
user@computer:/# php -v
# PHP 5.6.21-1+donate.sury.org~xenial+2 (cli)
# Copyright (c) 1997-2016 The PHP Group
# Zend Engine v2.6.0, Copyright (c) 1998-2016 Zend Technologies
# with Zend OPcache v7.0.6-dev, Copyright (c) 1999-2016, by Zend Technologies
```

From the commandline, I have both `php5.6` and `php7.2` available as commands. I also still have a `php` command - look in `/etc/alternatives` to see that it symlinks to a particular version of PHP cli\*. You can also quickly check which yours is using by running `php -v`.

Remember that `php.ini` and other files with settings are not in `/etc/php5` anymore but in `/etc/php/version/`

## 1.6. PHP Debugger

You may use a nice (but closed source) IDE such as PHPStorm (which provides a version at no cost for FLOSS projects such as Tiki), or stick with pure FLOSS apps such as Aptana Studio IDE.

See:

- Download:  
<http://www.apтана.com/products/studio3/download><sup>[6]</sup>
- Installation:  
<http://linuxpitstop.com/install-apтана-studio-3-on-centos-and-ubuntu/><sup>[7]</sup>
- Setup XDebug for PHP applications  
<http://digitaldiseny.com/es/blog/206-debug-php-apтана-ubuntu><sup>[8]</sup>  
<http://devcircle.blogspot.com.es/2013/02/enable-xdebug-in-apтана-studio-3x.html><sup>[9]</sup>
- Add svn integration  
<https://www.development-cycle.com/2013/09/installing-svn-support-in-apтана-studio-3/><sup>[10]</sup>  
<http://majadc.com/installing-subversion-plugin-in-apтана-studio-3><sup>[11]</sup>

## 1.7. Fix Shutter editor

Shutter seems to be missing some perl dependencies in repositories to have the edit feature installed by default. You can add packages by hand (they work fine) with:

### Commands in a terminal



```
cd ~/tmp
wget
https://launchpad.net/ubuntu/+archive/primary/+files/libgoocanvas-common_1.0.0-1_all
1.deb
wget
https://launchpad.net/ubuntu/+archive/primary/+files/libgoocanvas3_1.0.0-1_amd64.de
b
sudo dpkg -i libgoocanvas-common_1.0.0-1_all.deb
sudo dpkg -i libgoocanvas3_1.0.0-1_amd64.deb
wget
https://launchpad.net/ubuntu/+archive/primary/+files/libgoo-canvas-perl_0.06-2ubunt
u3_amd64.deb
sudo dpkg -i libgoo-canvas-perl_0.06-2ubuntu3_amd64.deb
sudo apt-get install -y libextutils-dependends-perl libextutils-pkgconfig-perl
sudo dpkg -i libgoo-canvas-perl_0.06-2ubuntu3_amd64.deb
sudo apt-get -f install
```

Then close shutter, and shutter indicator icon in the task bar, and restart it again. YOu'll be able to edit images again from within shutter.

## 1.8. Video Edition

The first program to try can be [OpenShot](#) (multiplatform).

But if you need more power/features, you can use **KDEnLive** (GNU/Linux only, and maybe MacOSX but not Windows).

And if you want to use a newer version than the one that comes with your distribution (either \*ubuntu 16.04, 18.04, etc., or even with the Kubuntu-ppa/Backports repo), you can install KDEnLive by means of the **Flatpack** system, which works nicely in my first tests



```
sudo add-apt-repository ppa:alexlarsson/flatpak
sudo apt update
sudo apt install flatpak # install flatpack software
sudo apt install gnome-software-plugin-flatpak # Install the Software Flatpak
plugin (optional)
flatpak remote-add --if-not-exists flathub
https://flathub.org/repo/flathub.flatpakrepo # -- Add the Flathub repository
flatpak install flathub org.kde.kdenlive # Install KdenLive through Flatpack -
KDEnLive version 18.12.0 by the time of this writing December 2018 ! on top of
Kubuntu 16.04
flatpak run org.kde.kdenlive # Run KdenLive through Flatpack
```

## 1.9. Dowload your own full YouTube Playlists

You can easily download your own videos of youtube (or even your entire playlsits) through the application **youtube-dl** or **yt-dlp**.

# 1. youtube-dl

You can install **youtube-dl** from repositories, even if it won't be updated as frequently as needed sometimes, to get sync with youtube way of working. Therefore, you are encouraged to install through the command-line:

### Commands in a terminal window



```
sudo apt update && sudo apt install youtube-dl # install programs from repos if you
don't have them yet
sudo wget https://yt-dl.org/downloads/latest/youtube-dl -O /usr/local/bin/youtube-
dl # get latest version from developers site
sudo chmod a+rx /usr/local/bin/youtube-dl # fix permissions of the downloaded file
sudo youtube-dl -U # update your previous youtube-dl version
```

You can download your own youtube playlist with:

#### Commands in a terminal window



```
youtube-dl -ci "https://www.youtube.com/playlist?list=YOURPLAYLISTID" -o  
"%(title)s-%(id)s.%(ext)s"
```

Replace **YOURPLAYLISTID** with your own YouTube playlist id (note it's not your youtube channel ID, but the youtube playlist id). You can get it from the url of a video when shown through it's own playlist. That url might look like:

- [https://www.youtube.com/watch?v=guFSSJTjyZI&list=\[12\]PLM2ch5TKp00dtMZeQCOJQw6BwJ19VDcj4&index=2](https://www.youtube.com/watch?v=guFSSJTjyZI&list=[12]PLM2ch5TKp00dtMZeQCOJQw6BwJ19VDcj4&index=2)
- [https://www.youtube.com/playlist?list=\[13\]PLM2ch5TKp00dtMZeQCOJQw6BwJ19VDcj4](https://www.youtube.com/playlist?list=[13]PLM2ch5TKp00dtMZeQCOJQw6BwJ19VDcj4)

Therefore, the playlist id is that part after **list=** and before any ampersand (&) if any.  
In this case: **PLM2ch5TKp00dtMZeQCOJQw6BwJ19VDcj4**

Therefore, this would be the instruction in the command line:

#### Commands in a terminal window



```
youtube-dl -ci  
"https://www.youtube.com/playlist?list=PLM2ch5TKp00dtMZeQCOJQw6BwJ19VDcj4" -o  
"%(title)s-%(id)s.%(ext)s"
```

And if you need in mp4 format only (because your tv doesn't accept some webm or mkv codec files, or whatever other reason), you can add these extra params **-f**

**'bestvideoext=mp4+bestaudioext=m4a/mp4'** :

#### Commands in a terminal window



```
youtube-dl -ci -o "%(title)s-%(id)s.%(ext)s" -f  
'bestvideo[ext=mp4]+bestaudio[ext=m4a]/mp4'  
"https://www.youtube.com/playlist?list=PLM2ch5TKp00dtMZeQCOJQw6BwJ19VDcj4"
```

# 1.1. Looking for a GUI for youtube-dl? youtube-dlg

There is a nice GUI called youtube-dlg, however, it doesn't seem to be updated as frequently as needed to keep it updated to latest youtube-dl changes (and therefore, with youtube changes).

You might try the GUI Download-DLG, in case it still works for you (if did for me months ago, but not currently, so I had to go to the command line to be able to fetch my own video playlist successfully)



```
#sudo add-apt-repository ppa:nilarimogard/webupd8 # needed once for youtube-dlg
sudo apt update && sudo apt install youtube-dlg # install programs from repos if
you don't have them yet
# refresh youtube-dl from the commands above so that you have the latest version
installed
```

Derived from:

- <http://www.webupd8.org/2014/03/multi-platform-youtube-dl-gui-youtube.html><sup>[14]</sup>

## 2. yt-dlp

If youtube-dl doesn't work for you any more (it did fail for me in 2023-10), you can try installing [yt-dlp](#) from discover app, or through pip, and launch the command with this program

### Commands in a terminal window



```
yt-dlp -ci -o "%(title)s-%(id)s.%(ext)s" -f
'bestvideo[ext=mp4]+bestaudio[ext=m4a]/mp4'
"https://www.youtube.com/playlist?list=PLM2ch5TKp00dtMZeqC0JQw6BwJ19VDcj4"
```

You can check the version of yt-dlp with:



```
$ yt-dlp -U
Current version: stable@2023.12.30 from yt-dlp/yt-dlp
Latest version: stable@2024.04.09 from yt-dlp/yt-dlp
```

If installed with pip, you will see this message:



```
ERROR: You installed yt-dlp with pip or using the wheel from PyPi; Use that to update
```

To upgrade you need to run a command out of two, it depends on how you installed it.

If you installed it via pip, use:



```
pip install --upgrade yt-dlp
```

otherwise use:



```
yt-dlp -U
```

Example:



```
pip install --upgrade yt-dlp
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: yt-dlp in /home/xavi/.local/lib/python3.10/site-packages (2023.12.30)
Collecting yt-dlp
  Downloading yt_dlp-2024.4.9-py3-none-any.whl (3.1 MB)
    ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 3.1/3.1 MB 391.7 kB/s eta 0:00:00
Requirement already satisfied: websockets>=12.0 in /home/xavi/.local/lib/python3.10/site-packages (from yt-dlp) (12.0)
Requirement already satisfied: pycryptodomex in /usr/lib/python3/dist-packages (from yt-dlp) (3.11.0)
Requirement already satisfied: brotli in /usr/lib/python3/dist-packages (from yt-dlp) (1.0.9)
Requirement already satisfied: requests<3,>=2.31.0 in /home/xavi/.local/lib/python3.10/site-packages (from yt-dlp) (2.31.0)
Requirement already satisfied: urllib3<3,>=1.26.17 in /home/xavi/.local/lib/python3.10/site-packages (from yt-dlp) (2.2.0)
Requirement already satisfied: mutagen in /home/xavi/.local/lib/python3.10/site-packages (from yt-dlp) (1.47.0)
Requirement already satisfied: certifi in /usr/lib/python3/dist-packages (from yt-
```

```

dlp) (2020.6.20)
Requirement already satisfied: charset-normalizer<4,>=2 in
/home/xavi/.local/lib/python3.10/site-packages (from requests<3,>=2.31.0->yt-dlp)
(3.3.2)
Requirement already satisfied: idna<4,>=2.5 in /usr/lib/python3/dist-packages (from
requests<3,>=2.31.0->yt-dlp) (3.3)
Installing collected packages: yt-dlp
  Attempting uninstall: yt-dlp
    Found existing installation: yt-dlp 2023.12.30
    Uninstalling yt-dlp-2023.12.30:
      Successfully uninstalled yt-dlp-2023.12.30
ERROR: pip's dependency resolver does not currently take into account all the
packages that are installed. This behaviour is the source of the following
dependency conflicts.
spotdl 4.2.4 requires yt-dlp<2024.0.0,>=2023.11.16, but you have yt-dlp 2024.4.9
which is incompatible.
Successfully installed yt-dlp-2024.4.9

```

You might also install sitewide with root permissions, but it is not recommended. See below:

“

***WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager. It is recommended to use a virtual environment instead: <https://pip.pypa.io/warnings/venv><sup>[15]</sup>***

Example run as root user:



```

sudo pip install --upgrade yt-dlp
[sudo] contrasenya per a xavi:
Collecting yt-dlp
  Downloading yt_dlp-2024.4.9-py3-none-any.whl (3.1 MB)
    _____ 3.1/3.1 MB 213.7 kB/s eta 0:00:00
Requirement already satisfied: brotli in /usr/lib/python3/dist-packages (from yt-
dlp) (1.0.9)
Collecting mutagen
  Downloading mutagen-1.47.0-py3-none-any.whl (194 kB)
    _____ 194.4/194.4 KB 565.0 kB/s eta 0:00:00
Requirement already satisfied: certifi in /usr/lib/python3/dist-packages (from yt-
dlp) (2020.6.20)
Collecting websockets>=12.0
  Downloading websockets-12.0-cp310-cp310-
manylinux_2_5_x86_64.manylinux1_x86_64.manylinux_2_17_x86_64.manylinux2014_x86_64.w
hl (130 kB)
    _____ 130.2/130.2 KB 398.3 kB/s eta 0:00:00

```



```

Collecting requests<3,>=2.31.0
  Downloading requests-2.31.0-py3-none-any.whl (62 kB)
    _____ 62.6/62.6 KB 363.6 kB/s eta 0:00:00
Collecting urllib3<3,>=1.26.17
  Downloading urllib3-2.2.1-py3-none-any.whl (121 kB)
    _____ 121.1/121.1 KB 599.6 kB/s eta 0:00:00
Requirement already satisfied: pycryptodomex in /usr/lib/python3/dist-packages
(from yt-dlp) (3.11.0)
Requirement already satisfied: idna<4,>=2.5 in /usr/lib/python3/dist-packages (from
requests<3,>=2.31.0->yt-dlp) (3.3)
Collecting charset-normalizer<4,>=2
  Downloading charset_normalizer-3.3.2-cp310-cp310-
manylinux_2_17_x86_64.manylinux2014_x86_64.whl (142 kB)
    _____ 142.1/142.1 KB 553.8 kB/s eta 0:00:00
Installing collected packages: websockets, urllib3, mutagen, charset-normalizer,
requests, yt-dlp
  Attempting uninstall: urllib3
    Found existing installation: urllib3 1.26.5
    Not uninstalling urllib3 at /usr/lib/python3/dist-packages, outside environment
/usr
    Can't uninstall 'urllib3'. No files were found to uninstall.
  Attempting uninstall: requests
    Found existing installation: requests 2.25.1
    Not uninstalling requests at /usr/lib/python3/dist-packages, outside
environment /usr
    Can't uninstall 'requests'. No files were found to uninstall.
Successfully installed charset-normalizer-3.3.2 mutagen-1.47.0 requests-2.31.0
urllib3-2.2.1 websockets-12.0 yt-dlp-2024.4.9
WARNING: Running pip as the 'root' user can result in broken permissions and
conflicting behaviour with the system package manager. It is recommended to use a
virtual environment instead: https://pip.pypa.io/warnings/venv

```

Derived from:

- <https://forum.peppermintos.com/index.php?topic=5303.0><sup>[16]</sup>
- <https://unix.stackexchange.com/a/328536><sup>[17]</sup>

Alias names for this page:

playlist | playlists | Youtube | YoutubeLists | YoutubeList | Youtube Lists | Youtube List

## 2.1. Make Customized Ubuntu Live USB

See **UCK**: **Ubuntu Customization Kit**, which is still available in the default ubuntu repositories in 16.04 and 18.04 LTS.

<https://www.makeuseof.com/tag/ubuntu-customization-kit-linux-operating-system/><sup>[18]</sup>

If you want to attempt to make Live USB with persistent changes, you can try making the usb with mksub

(see below: #Create\_Linux\_Live\_USB\_disks)

## 2.1.1. Example for a OMDbuntu Live USB disk

Choose Lubuntu 18.04 64 bit as a base (it doesn't seem to work as expected for Lubuntu 16.04 since it doesn't seem to be able to find internet repos to get packages from).

Desktop environment: Other

Lang: ca

Please choose customization action

1) Run console application

2) Continue building

Your choice (1..2)> 1

Extra system packages:

### Extra system packages installed in a terminal window inside UCK



```
add-apt-repository -y ppa:webupd8team/java
add-apt-repository -y ppa:marutter/rrutter
apt install -y gfxboot-theme-ubuntu squashfs-tools syslinux-utils libnss-winbind
winbind cifs-utils oracle-java8-installer nfs-common git git-cola mc htop
parcellite kupfer p7zip-full printer-driver-cups-pdf meld sendmail gigolo
apt install -y r-recommended r-cran-xml libgraphviz-dev libcairo2-dev r-cran-
cairodevice freeglut3 freeglut3-dev r-cran-rgl r-cran-rgl r-cran-misc3d libx11-
dev libxt-dev libcurl4-gnutls-dev libxml2-dev r-cran-xml libgraphviz-dev libcairo2-
dev bwidget tk-table libv8-dev r-cran-rjava libmpfr-dev libc6 libssl-dev texlive-
latex-extra texlive-lang-spanish libx11-dev libxml2-dev libxt-dev r-cran-misc3d
subversion git texmaker tk-dev unaccent xfb libgdal-dev libproj-dev libmagick++-
dev r-cran-rcolorbrewer r-cran-doparallel libssh2-1-dev libudunits2-dev libgdal-dev
libgeos-dev libproj-dev libjq-dev libprotobuf-dev protobuf-compiler libssl-dev
libcairo2-dev libapparmor-dev g++
# apt install -y r-cran-mysql # no troba el paquet slax a debian9
R CMD javareconf
wget https://download1.rstudio.org/rstudio-xenial-1.1.463-amd64.deb
apt install -y ./rstudio-xenial-1.1.463-amd64.deb
```

Extra R packages

### Extra R packages installed in a terminal window inside UCK



```
if (!require(devtools)) install.packages("devtools"); library(devtools)
# Check Availability of package manager "PacMan" (and install it, if missing)
```

```
# if you have R < 3.5
if (version$major==3 && version$minor < 5) {
  # Instal·la devtools si et cal
  if (!require("devtools")) install.packages("devtools"); require("devtools")
  cat("R Version: ", paste0(version$major,".",version$minor), ". ")
  cat("Tens una versió de R anterior a 3.5, per tant, instal·lem la versió
corresponent de PacMan (0.4.1)\n")
  # Instal·la la darrera versió de pacman (0.4.1) que anava abans de la que va
només amb R 3.5+ (pacman v0.5)
  if (!require("pacman")) install_version("pacman", version = "0.4.1");
  require("pacman")
} else {
# if you have R 3.5 or higher
  if (!require("pacman")) install.packages("pacman"); require("pacman")
}
p_load("devtools", "plotly", "Nozzle.R1", "VennDiagram", "stringr", "xml2",
"parallel", "data.table", "doParallel", "xtable", "plyr", "dplyr", "reshape2",
"rjson", "d3heatmap", "htmlwidgets", "googleVis", "doMC", "knitr", "checkpoint",
"sem", "rmarkdown", "relimp", "aplpack", "addinslist", "tidyverse", "magick",
"webshot", "pacman", "officer", "flextable", "huxtable", "RODBC", "leaflet",
"gitlabr", "colorspace", "mapdeck", "rsvg", "png", "cartography", "magick",
"rJava", "rio")
```

Before exiting the "-+Run console application+-", you need to update (or replace) the customization script in order to make it work with latest Ubuntu versions (16.04+; derived from here<sup>[19]</sup>)

Create a file called `customize_iso2` in your home directory for instance:



```
nano /home/xavi/customize_iso2
```

Add these contents inside:

### Contents of `customize_iso2`



```
#!/bin/bash

#####
# UCK - Ubuntu Customization Kit                                     #
# Copyright (C) 2006-2010 UCK Team                                   #
#                                                                     #
# UCK is free software: you can redistribute it and/or modify      #
# it under the terms of the GNU General Public License as published by #
# the Free Software Foundation, either version 3 of the License, or #
# (at your option) any later version.                               #
```

```

#                                                                                                     #
# UCK is distributed in the hope that it will be useful,                                           #
# but WITHOUT ANY WARRANTY; without even the implied warranty of                                   #
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the                                     #
# GNU General Public License for more details.                                                       #
#                                                                                                     #
# You should have received a copy of the GNU General Public License                                 #
# along with UCK.  If not, see <http://www.gnu.org/licenses/>.                               #
#####

# NAME:
#   customize_iso -- customize iso image outside of root FS
#
# SYNOPSIS:
#   customize_iso [remaster_home] [customization_scripts_dir]
#
# DESCRIPTION:
#   This procedure performs additional customization that needs to
#   happen outside of the image of the root file system:
#       - Configuration of the boot environment
#           - isolinux configuration
#           - gfxboot configuration (language/keyboard support!)
#           - propagation of a new kernel/initrd
#           - Simple bootmanager (directory "install" in image)
#       - Configuration of additional resources outside the root FS
#           - Add-Ons (like sample document/music/video files)
#           - More Add-Ons (like OSS for Win / Mac)
#
# NOTES:
#   Execution of the procedure may require Internet access to download
#   the source of the latest version gfxboot-theme-ubuntu.
#
#   If a local copy of gfxboot-theme-ubuntu is available in the remaster
#   home, it will be used instead of a (possibly updated) download.

SCRIPT_DIR=`dirname "$0"`
REMASTER_HOME=${1:-~/tmp}
SCRIPT_DIR=${2:-$REMASTER_HOME/customization-scripts}
ISO_REMASTER_DIR="$REMASTER_HOME/remaster-iso"
REMASTER_DIR="$REMASTER_HOME/remaster-root"
BOOT_LANG=`cat "$SCRIPT_DIR/livedcd_locale"`

function failure()
{
    echo "$@"
    exit 1
}

function get_latest_kernel()

```

```

{
    set -- $(ls "$REMASTER_DIR"/boot/vmlinuz* 2>/dev/null |
        sed -e "s@$REMASTER_DIR/boot/@" |
        tr -- '-.' '\t' |
        sort --key=2 --key=3 --key=4 --key=5 --numeric-sort |
        tail -n1 )
    [ "$1" = "vmlinuz" ] && echo "$REMASTER_DIR/boot/$1-$2.$3.$4-$5-$6"
}

# Create a temporary directory to assemble the gfxboot stuff in
BUILD_DIR=`mktemp -d`
if [ -d $REMASTER_HOME/gfxboot-theme-ubuntu ]
then
    cp -r $REMASTER_HOME/gfxboot-theme-ubuntu "$BUILD_DIR" ||
        failure "Cannot copy gfxboot-theme-ubuntu to $BUILD_DIR"
    pushd "$BUILD_DIR" >/dev/null ||
        failure "Cannot change directory to $BUILD_DIR"
else
    pushd "$BUILD_DIR" >/dev/null ||
        failure "Cannot change directory to $BUILD_DIR"
    #-----HS (1)-->
    #DISTRO_CODENAME=`cd "$ISO_REMASTER_DIR"/dists && find . -maxdepth 1 -type d |
grep '/' | cut -d '/' -f2` ||
    # failure "Unable to identify Ubuntu distro codename"
    DISTRO_CODENAME="$(lsb_release -c|awk '{print $2}')"||
        failure "Unable to identify Ubuntu distro codename"
    #-----HS (1)--<
    APT_SOURCES_TMP_DIR=`mktemp -d`
    wget -c
    http://archive.ubuntu.com/ubuntu/ubuntu/ubuntu/dists/$DISTRO_CODENAME/main/source/S
ources.gz -O "$APT_SOURCES_TMP_DIR"/Sources.gz
    #-----HS (2)-->
    #GFXBOOT_THEME_UBUNTU_SOURCE_PACKAGE=http://archive.ubuntu.com/ubuntu/ubuntu/ubuntu
/pool/main/g/gfxboot-theme-ubuntu/$(zgrep gz "$APT_SOURCES_TMP_DIR"/Sources.gz |
grep gfxboot-theme-ubuntu | sed -n 1p | awk '{ print $3 }')
    GFXBOOT_THEME_UBUNTU_SOURCE_PACKAGE=http://archive.ubuntu.com/ubuntu/ubuntu/ubuntu/
pool/main/g/gfxboot-theme-ubuntu/$(dpkg -l|grep gfxboot-theme-ubuntu|awk '{print
$2_"$3_"$4".deb"}')
    #-----HS (2)--<
    wget $GFXBOOT_THEME_UBUNTU_SOURCE_PACKAGE ||
        failure "Unable to download gfxboot-theme-ubuntu source package from
$GFXBOOT_THEME_UBUNTU_SOURCE_PACKAGE"
    #-----HS (3)-->
    #tar xzf *.tar.gz ||
    tar xzf *.tar.gz
    dpkg -x *.deb . ||
        failure "Unable to extract gfxboot-theme-ubuntu source package"
    #-----HS (3)--<
fi

```

```

# Build the gfx boot theme
#-----HS (4)-->
cd usr/share
#-----HS (4)--<
cd gfxboot-theme-ubuntu
cd po
ln -s pt.po pt_PT.po
cd ..
##if [ "$BOOT_LANG" = "pt_PT" ]; then
## make DEFAULT_LANG="pt" || failure "Failed to build gfxboot theme"
##else
# make DEFAULT_LANG="$BOOT_LANG" || failure "Failed to build gfxboot theme"
##fi

# Fix list of languages
pushd boot >/dev/null

# Create regexp that matches all language packs on CD
langpack=""
for langpack in `cat "$SCRIPT_DIR/language_packs"`; do
    if [ -z "$LANGPACKS" ]; then
        LANGPACKS="$langpack"
    else
        LANGPACKS="$LANGPACKS|$langpack"
    fi
done

# Rewrite langlist
cat "$SCRIPT_DIR/livecd_locales" >langlist
popd >/dev/null

# Copy to isolinux in image directory
cp -af boot/* "$ISO_REMASTER_DIR/isolinux/" ||
    failure "Error while copying boot files to $ISO_REMASTER_DIR/isolinux"

popd >/dev/null

# Cleanup
[ "$BUILD_DIR" != "/" ] && rm -rf "$BUILD_DIR"

# Copy kernel and initrd, in case it was changed during installation
VMLINUZ=$(get_latest_kernel)
if [ "$VMLINUZ" != "" ]
then
    INITRD="$REMASTER_DIR/boot/initrd.img-$(echo `basename $VMLINUZ` | cut -d'-' -f 2-)"
    if [ -e "$VMLINUZ" -a -e "$INITRD" ]
    then
        echo "Updating kernel:"
    fi
fi

```

```

    echo "    kernel=$VMLINUZ"
    echo "    initrd=$INITRD"
    cp -f "$VMLINUZ" "$ISO_REMASTER_DIR/casper/vmlinuz"
    cp -f "$INITRD" "$ISO_REMASTER_DIR/casper/initrd.gz"
else
    echo "Not updating kernel as initrd not present"
fi
fi

# Misc fixup for Karmic
pushd "$ISO_REMASTER_DIR"/isolinux >/dev/null

# What used to be called "message" is now called "bootlogo"
if [ -f isolinux.cfg -a -n "`grep "gfxboot bootlogo" isolinux.cfg 2>/dev/null`" ]
then
    if [ -f message ]
    then
        echo "Using bootlogo instead of message"
        mv message bootlogo
    fi
fi

# What used to be a gzipped initrd now is a lzma compressed initrd
if [ -f text.cfg ] || [ -f txt.cfg ]
then
    # At least one of the .cfg file will be missing. Drop error message.
    lzused=`grep initrd.lz text.cfg txt.cfg 2>/dev/null`
    lzmacmd=`which lzma`
    if [ -n "$lzused" ]
    then
        if [ -n "$lzmacmd" ]
        then
            if [ -f ../casper/initrd.gz ]
            then
                pushd "$ISO_REMASTER_DIR"/casper >/dev/null
                echo "Recompressing initrd from gzip to lzma"
                rm -f initrd.lz
                gunzip <initrd.gz | $lzmacmd >initrd.lz
                rm -f initrd.gz
                popd >/dev/null
            fi
        else
            if [ -f ../casper/initrd.gz ]
            then
                echo "lzma command not installed"
                echo "Switching permanently to gzipped initrd"
                sed -i -e 's/initrd\.lz/initrd.gz/g' text.cfg txt.cfg
                rm -f ../casper/initrd.lz
            else

```

```
        : do nothing - no initrd.gz
    fi
fi
else
    : do nothing - initrd.lz not used
fi
fi
```

Then overwrite the `customize_iso` in `uck` with your own one:

**Command to be run on a terminal window out of UCK**



```
cp /home/xavi/customize_iso2 /home/xavi/tmp/customization-scripts/customize_iso
```

Then you can continue with step [2](#) in:

“

*Please choose customization action*

*1) Run console application*

*2) Continue building*

*Your choice (1..2) 2*

## 2.2. Create Linux Live USB disks

### 2.2.1. With mkusb

You can add an extra repository to include [mkusb dus](#) (**Do USB Stuff**), which will allow you to create boot usb disks with persistence of changes in a reliable way for recent Ubuntu distros such as 16.04 LTS and newer. In such distros the default traditional methods such as USB Disk Creator or UnetBootIn no longer work in all cases as they did with earlier Ubuntu versions.

Derived from: <https://help.ubuntu.com/community/mkusb><sup>[20]</sup>



```
sudo add-apt-repository ppa:mkusb/ppa # and press Enter
sudo apt-get update
sudo apt-get install mkusb mkusb-nox usb-pack-efi
```



And system for a tweaked version of LXLE 16.04.2 (32 bits; 1-disk-fits-all-computers approach):  
<https://help.ubuntu.com/community/mkusb/persistent/LXLE><sup>[21]</sup>

But there are many other pre-built images, such as Lubuntu 16.04 32bits and 64 bits:  
<https://help.ubuntu.com/community/mkusb/persistent><sup>[22]</sup>  
<https://help.ubuntu.com/community/mkusb/persistent/lubuntu><sup>[23]</sup>

## 2.2.2. With Slax

- Get a 16Gb USB disk
- Make two partitions  
leave 4Mb of free space at the beginning of usb disk
  1. ~8.5Gb ext4 or ext3
  2. ~7.5Gb ntfs
- uncompress slax.iso in your linux disk (with `isomaster` for instance)
- copy your slax folder to your usb ext4 partition
- run as root the file in your usb ext4 partition: `sudo bash ./slax/boot/bootinst.sh` to make your usb bootable
- shutdown your computer, and boot it again this time from your usb stick. Slax should boot
- make your desired changes following guidelines on slax.org website
- change keyboard by hand tweaking a file in the persistent changes folder
  - Edit this file `/root/.fluxbox/startup` and add there near the top a line such as



```
setxkbmap es
```

- reboot to check that your changes were kept as desired
- clone your usbdisk (with partitions and all) with dd.
  - find our your usb disk device name (sdc in my case). you can use command `sudo lsblk` or `sudo fdisk -l`
  - make image with dd. Example:



```
sudo dd if=/dev/sdc of=/media/xavi/mollero_ext4/tmp/backup_slaxomdr.img  
bs=4M status=progress
```

- clone back over another usb with dd again (or theoretically with a dd GUI, but gdiskdump fails for me on Kubuntu 16.04)  
To restore the image to another USB drive just invert the process. Just make sure that the new USB drive (/dev/sdd in this example) is as big or bigger than the original one



```
sudo dd if=/media/xavi/mollero_ext4/tmp/backup_slaxomdr.img of=/dev/sdd  
bs=4M status=progress
```

## 2.3. Mapes mentals

Una aplicació de KDE que sembla molt recomanada és **Semantik**. Respecte a les aplicacions multiplataforma, les més recomanades serien FreeMind, o Xmind.

### 2.3.1. Semantik

Veure: <https://waf.io/semantik.html><sup>[24]</sup>

Features:

- Open-source with source code on GitHub and available for download here.
- Mind maps are either in tree or 2-dimension form.
- Create UML diagrams, flowcharts, technical documentation, etc.
- Generate “flat” documents such as reports and presentations.
- Supports LaTeX, OpenOffice, and HTML document formats.
- Export diagrams in PDF or PNG format.
- Command line tools for document exports.

Essential commands with Semantik:

- Double click to add a concept, type "Enter" to edit it
- Double click on a leaf or use the link mode to connect concepts
- Double click on a link to remove it
- Middle button + move to scroll on the map
- Scroll wheel for zooming
- When a concept is selected, press enter to start editing it
- Control+R to reorganize the map
- Control+H to center the zoom on selected objects
- When a map is ready, use Ctrl+G to generate a document

### 2.3.2. XMind

Hi ha un projecte en github d'un instal·lador en bash

(<https://github.com/mriza/XMind-Linux-Installer><sup>[25]</sup>), al que cal fer només una petita acció posterior per a que tot funcioni (al menys en el Kubuntu 18.04 que jo he provat i documentat ). Cal eliminar el contingut d'aquesta línia:

<https://github.com/mriza/XMind-Linux-Installer/blob/master/xmind-installer.sh#L81><sup>[26]</sup>



```
--add-modules=java.se.ee
```

de l'arxiu que genera a `/opt/xmind/XMind_amd64/XMind.ini`

Passos a seguir:

1. Cal tenir instal·lats un paquets de sistema base (que en teoria ja els tens si has seguit els passos anteriors):



```
sudo apt-get install -y unzip default-jre libgtk2.0-0 libwebkitgtk-1.0-0 lame  
libc6 libglib2.0-0
```

2. Baixat el projecte de codi via git:

**Instrucció en terminal de Linux**



```
git clone https://github.com/mriza/XMind-Linux-Installer.git
```

3. Download XMind from <http://www.xmind.net/download/linux/><sup>[27]</sup> (la versió 8 ja va bé)
4. Add execute permission to the installer script and run it,

**Instrucció en terminal de Linux**



```
chmod +x xmind-installer.sh; sudo ./xmind-installer.sh elteusuariubuntu
```

Edita l'arxiu `/opt/xmind/XMind_amd64/XMind.ini` per eliminar aquesta línia:



```
--add-modules=java.se.ee
```

I ja hauria d'estar. Pots obrir XMind amb `Alt` + `Barra Espaiadora` i escriure **XMind**.

A mi m'ha funcionat amb Java 1.8.

## 2.4. WPS-Office

WPS Office is an office productivity suite, including Writer, Presentation and Spreadsheets. WPS Office is a native snap package in Ubuntu. To install it in all currently supported versions of Ubuntu open the terminal and type:



```
# Option 1
sudo snap install wps-office-all-lang-no-internet # all interface languages & check
spell tools and without internet connection to their servers

## Option 2
#sudo snap install wps-office-multilang # all interface languages and with internet
connection to their servers

## Option 3
#sudo snap install wps-office # only english interface and with internet connection
to their servers
```

The wps-office snap package will be updated automatically when updates are available

This is a multilanguage snap package for wps-office. It contains all interface languages available for the Windows version. This package is based on the wps-office-all-lang-no-internet package and the official wps-office package. The official wps-office can be found at <https://snapcraft.io/wps-office>.<sup>[28]</sup>

The package wps-office-all-lang-no-internet can be found at

<https://snapcraft.io/wps-office-all-lang-no-internet><sup>[29]</sup> or

<https://github.com/cyrpaut/wps-office-all-lang-no-internet>.<sup>[30]</sup> Because of security and privacy this package is shipped without internet plug. So there is no risk of a backdoor.

This package can be found at <https://snapcraft.io/wps-office-multilang><sup>[31]</sup> and <https://github.com/HoLuLuLu/wps-office-multilang>.<sup>[32]</sup>

## 2.5. Element Secure Messener

A new world of messaging and collaboration

No datamining - Own your conversations, choose where they live

No walled gardens - Talk to anyone, whatever app they are using

No eavesdropping - Be sure you're talking to the right people, in private

Typical messaging apps own your conversations, mine your data and lock you into a walled garden.

Element is not a typical messaging app.

Element is unique. You own your data and decide where it lives. Element lets you talk securely to anyone, regardless of their app.

Personal use - 20M+ users worldwide

Communities - Communities from fan clubs to Firefox

Business - 2M+ business users

Public sector - Trusted by French, US and German governments

<https://element.io/get-started><sup>[33]</sup>



```
sudo apt install -y wget apt-transport-https
```

```
sudo wget -O /usr/share/keyrings/riot-im-archive-keyring.gpg
https://packages.riot.im/debian/riot-im-archive-keyring.gpg
echo "deb [signed-by=/usr/share/keyrings/riot-im-archive-keyring.gpg]
https://packages.riot.im/debian/ default main" | sudo tee
/etc/apt/sources.list.d/riot-im.list
sudo apt update
sudo apt install element-desktop
```

# Pending to be reviewed

[+]

<sup>[1]</sup> <https://linuxconfig.org/how-to-reset-root-mysql-password-on-ubuntu-18-04-bionic-beaver-linux>

<sup>[2]</sup> <https://svn.r-project.org/R/branches/ALTREP/ALTREP.html>

<sup>[3]</sup> <https://help.ubuntu.com/community/Boot-Repair>

<sup>[4]</sup> <http://www.webupd8.org/2014/02/modify-pdf-files-in-linux-with-master.html>

<sup>[5]</sup> <http://code-industry.net/free-pdf-editor.php>

<sup>[6]</sup> <http://www.apтана.com/products/studio3/download>

<sup>[7]</sup> <http://linuxpitstop.com/install-aptana-studio-3-on-centos-and-ubuntu/>

<sup>[8]</sup> <http://digitaldiseny.com/es/blog/206-debug-php-aptana-ubuntu>

<sup>[9]</sup> <http://devcircle.blogspot.com.es/2013/02/enable-xdebug-in-aptana-studio-3x.html>

<sup>[10]</sup> <https://www.development-cycle.com/2013/09/installing-svn-support-in-aptana-studio-3/>

<sup>[11]</sup> <http://majadc.com/installing-subversion-plugin-in-aptana-studio-3>

<sup>[12]</sup> <https://www.youtube.com/watch?v=guFSSJTjyZI&list=>

<sup>[13]</sup> <https://www.youtube.com/playlist?list=>

<sup>[14]</sup> <http://www.webupd8.org/2014/03/multi-platform-youtube-dl-gui-youtube.html>

<sup>[15]</sup> <https://pip.pypa.io/warnings/venv>

<sup>[16]</sup> <https://forum.peppermintos.com/index.php?topic=5303.0>

<sup>[17]</sup> <https://unix.stackexchange.com/a/328536>

<sup>[18]</sup> <https://www.makeuseof.com/tag/ubuntu-customization-kit-linux-operating-system/>

<sup>[19]</sup> <https://askubuntu.com/a/699474>

<sup>[20]</sup> <https://help.ubuntu.com/community/mkusb>

<sup>[21]</sup> <https://help.ubuntu.com/community/mkusb/persistent/LXLE>

<sup>[22]</sup> <https://help.ubuntu.com/community/mkusb/persistent>

<sup>[23]</sup> <https://help.ubuntu.com/community/mkusb/persistent/lubuntu>

<sup>[24]</sup> <https://waf.io/semantik.html>

<sup>[25]</sup> <https://github.com/mriza/XMind-Linux-Installer>

<sup>[26]</sup> <https://github.com/mriza/XMind-Linux-Installer/blob/master/xmind-installer.sh#L81>

<sup>[27]</sup> <http://www.xmind.net/download/linux/>

<sup>[28]</sup> <https://snapcraft.io/wps-office>.

<sup>[29]</sup> <https://snapcraft.io/wps-office-all-lang-no-internet>

<sup>[30]</sup> <https://github.com/cyrpaut/wps-office-all-lang-no-internet>.

- [31] <https://snapcraft.io/wps-office-multilang>
- [32] <https://github.com/HoLuLuLu/wps-office-multilang>.
- [33] <https://element.io/get-started>
- [34] <http://ubuntuforums.org/showthread.php?t=2314905>
- [35] <https://www.youtube.com/watch?v=l8xa-PkDhhk>
- [36] <http://www.bio.brandeis.edu/~shaikh/linux/suspend.htm>
- [37] <http://ubunlog.com/ya-disponible-mate-1-12-1-para-ubuntu-15-10-y-16-04-lts/>
- [38] <http://get.adobe.com/reader/otherversions/>
- [39] [http://ardownload.adobe.com/pub/adobe/reader/unix/9.x/9.5.5/enu/AdbeRdr9.5.5-1\\_i386linux\\_enu.deb](http://ardownload.adobe.com/pub/adobe/reader/unix/9.x/9.5.5/enu/AdbeRdr9.5.5-1_i386linux_enu.deb)
- [40] <https://sourceforge.net/projects/portablesigner/>
- [41] <http://www.sinadura.net/es/inicio>
- [42] <http://guimi.net/blogs/hiparco/firma-digital-con-debian-gnulinix-y-sinadura/>
- [43] <https://www.elastic.co/guide/en/elasticsearch/reference/5.0/deb.html#deb-repo>
- [44] <https://www.freedesktop.org/software/systemd/man/journalctl.html>
- [45] <https://www.elastic.co/guide/en/kibana/5.0/install.html>
- [46] <https://www.elastic.co/guide/en/kibana/5.0/deb.html>
- [47] <http://wiki.x2go.org/doku.php/doc:de-compat>
- [48] <https://www.howtoforge.com/tutorial/multiple-usb-writing-on-linux/>
- [49] <https://itsfoss.com/multiple-linux-one-usb/>
- [50] <http://multibootusb.org/>
- [51] [https://github.com/mbusb/multibootusb/releases/download/v9.2.0/python3-multibootusb\\_9.2.0-1\\_all.deb](https://github.com/mbusb/multibootusb/releases/download/v9.2.0/python3-multibootusb_9.2.0-1_all.deb)
- [52] <http://www.ubuntugeek.com/ddrescue-gui-a-python-script-to-make-it-easier-to-use-ddrescue.html>
- [53] <https://humdi.net/vnstat/>
- [54] <https://github.com/phw/peek>
- [55] <https://www.genbeta.com/paso-a-paso/como-usar-qbittorrent-para-buscar-descargar-torrents-tener-que-visitar-ningun-sitio-web>