MyLiveUSB2

This manual is divided into sections, each of which deal with individual issues, and includes snapshot images where appropriate.

We hope you enjoy using this utility and find this manual helpful in making the best use of it for your purposes.

Table of Contents

General Information

Some general information about drives and partitions is probably needed first.
For the purposes of this utility, your USB flash stick/drive/pendrive/whatever can be regarded in a similar manner to a Hard Disk Drive. It can be booted and partitioned in the same way.

There are two partitioning schemes which concern us. The first is msdos/MBR (Master Boot Record) which is the traditional partitioning scheme for Windows, Linux and other operating systems. There is a second partitioning scheme called GPT which was devised to overcome the limitations of MBR because HDDs grew much larger than envisaged by the MBR authors.

This utility is designed to deal with both schemes. So, whether you have your USB stick set up using MBR partitioning or GPT partitioning, this utility will perform in similar fashion.

A further development which had an impact on LiveUSB creation was the spread of UEFI firmware in PCs. This was developed to replace the BIOS and improve greatly on it. In that regard it does very well, providing the user with a more graphical experience and the ability to navigate it using a mouse.

Legacy booting requires an extra partition (bios_grub) on GPT drives but has no extra requirement on MBR drives. UEFI booting requires a specific partition (ESP) on the Drive used to boot that PC. This applies to both MBR and GPT drives.

As most operating systems are now UEFI compliant they get installed with UEFI enabled, and to continue to work, UEFI must be left enabled.

This impacts on LiveUSB creation, because such a tool needs to be able to boot regardless the BIOS or UEFI firmware, and furthermore should be able to install the OS from that live session. To help with this situation this utility now has the ability to set up both a Legacy (BIOS) and a UEFI bootable USB stick on the one device - providing the partition requirements are complied with regarding boot partition etc..

We have included a ‘How-To’ document explaining what is required when partitioning your USB device to enable either or both booting modes.

Note: Both booting schemes will be available on a GPT or MBR partitioned device provided the required partitions are present.

How to Prepare your USB Drive for Best Use

Be certain you select the correct drive for these operations as the drive contents will be lost forever.

Your first decision concerns the partitioning structure you prefer to use on the USB device. This can be MBR or GPT, and your decision will (as previously explained) determine what partitions are required.

Partition Table

You can launch Gparted from command line, specifying one drive, so that all others are excluded.
This will ensure no costly errors can happen to the other attached drives. First determine the drive you wish to work on .... let us say it is /dev/sdp .... and then issue the following command

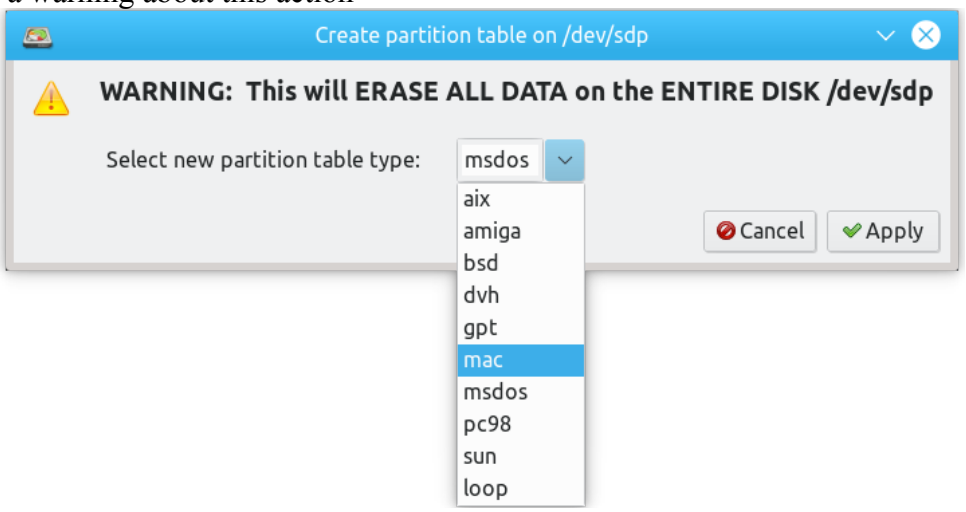
gparted /dev/sdp

If you did not use the command line to open Gparted, use the drop-down list to the right hand side of the Gparted window, to select the ***correct drive*** to work on.

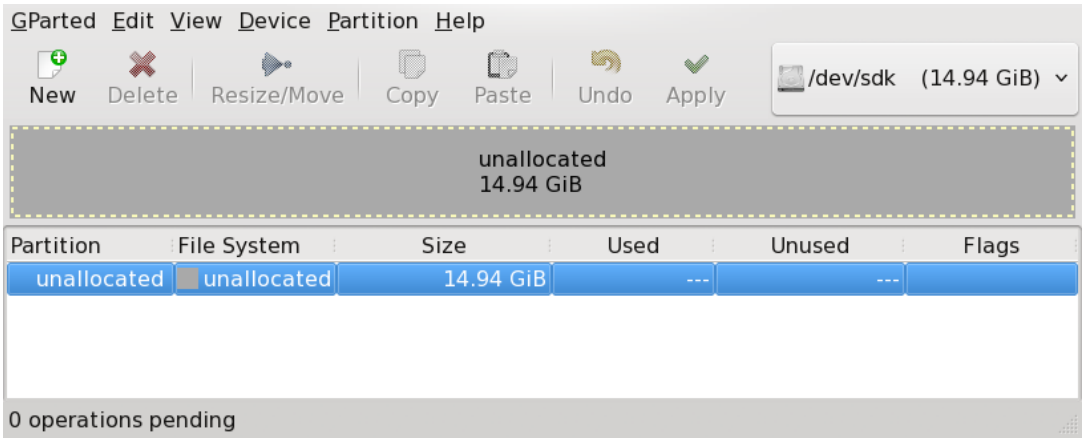
If in doubt, withdraw the USB device, refresh the Gparted device list (menu Gparted – Refresh Devices) and note the devices listed. Then re-insert the USB device and refresh the list. You should now be easily able to identify the extra device in the list.

In the menu, select 'Device - Create a Partition Table' – and then select **GPT** **or** **MBR** type from the available types.

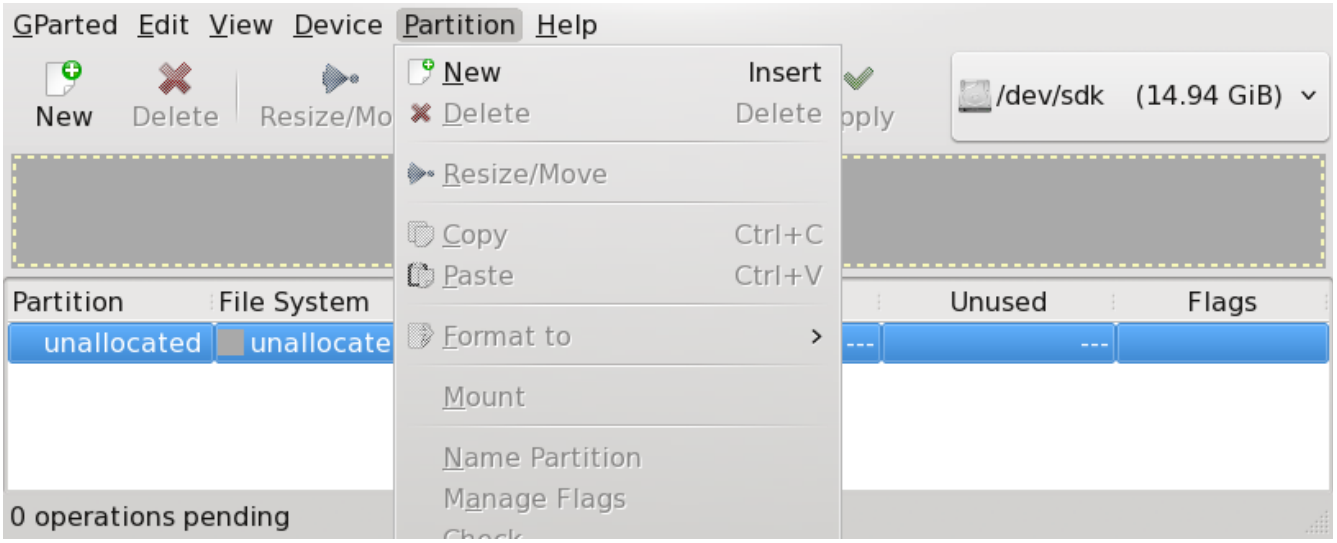
You will get a warning about this action



The window will now show a device with no partitions :-



Highlighting this empty space, we now select to create new partitions .... Menu – Partition – New and create the first partition we need, which is dependent on what we want to achieve and the partition table chosen.



This list provides the partition requirements for the various functions:-

	Goal	Partition Table Type	Partitions Required
•	Legacy boot	MBR	ext4
•	Legacy boot	GPT	bios_grub; ext4
•	EFI boot	MBR	ESP; ext4
•	EFI boot	GPT	ESP; ext4
•	EFI + Legacy boot	MBR	ESP; ext4
•	EFI + Legacy boot	GPT	ESP; bios_grub; ext4

Regardless which Partition Table type you chose the partitions are created in the same manner.

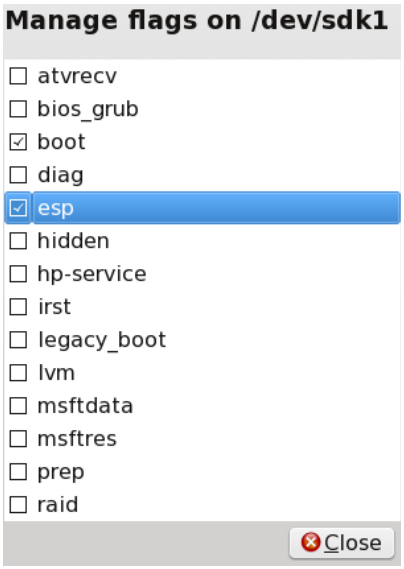
bios_grub partition

This is a very small partition (only 1MB in size). It is normally placed at the beginning of the device. When selecting what format to apply to it, scroll down the list and select ***unformatted***. When the change is applied, immediately go to Partitions – Flags and apply the ***bios_grub*** flag to it.

That is it. Nothing further is required. After the flag has been set the Gparted window should correctly reflect the type of partition it is. (see pic further down)

esp partition

You create this partition with size of 33MBs, and format it to FAT32. When created you then apply the esp flag and the Gparted display will show the changes made. When the esp flag is selected, the boot flag is automatically selected also.



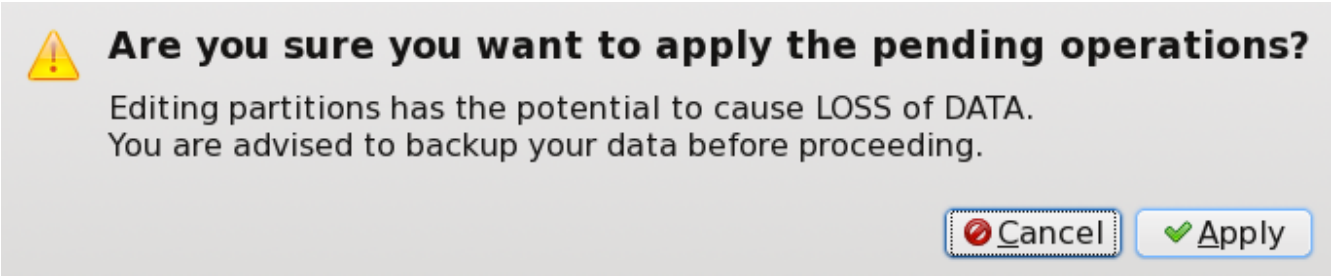
ext4 partition

It is usual to create the ext4 partition to include the rest of the space available on the device (particularly when using a small device). You can make this partition whatever size suits your needs. Just make sure it is large enough to accommodate the ISOs you intend to install there. Most ISO are greater than 1GB these days, so for 4 ISO the ext4 partition would need to be ~5GB or greater in size.
If you intend to use *Persistence* when booting the USB device then you need to make allowance for this also in the size selected for the ext4 partition.

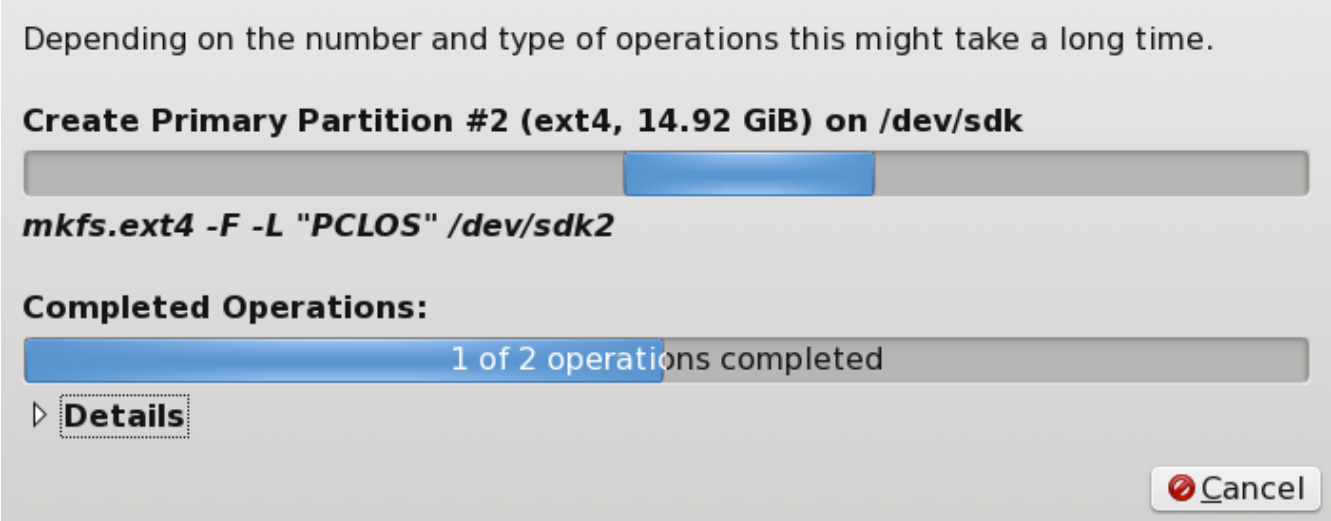
When using Gparted your changes are stored and the display changes to reflect what you ***intend*** to happen. Your changes are not applied until you deliberately do so.

I suggest you apply your changes after you have set up each partition, so you can also apply the flag when required.

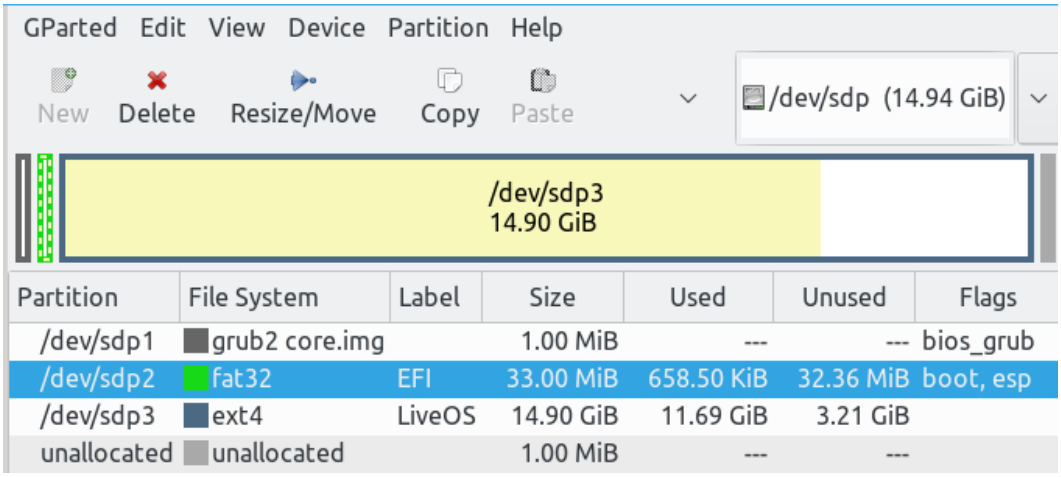
As you can see in this pic when you *Apply* you need to confirm by clicking Apply in this window.



You will get a progress window as the program works



When finished you should see partitions something like this.



What It Can Do

The MyLiveUSB2 utility is designed to do the following:-

- Create a LiveUSB of a PCLinuxOS ISO. It does not function with other ISOs.
- The ISO used can be an official release; a community release; or a personal remaster created with mylivecd or MyLiveGTK (which is a GUI front end to mylivecd).

It allows the user to place any number of different PCLOS ISOs on the one USB device (within its capacity of course), and refuses to use the same name twice to avoid confusion.

This is one of the biggest attractions of this utility, as it allows a user to carry with them several versions of PCLOS on one USB stick, for instance:

- A remaster of your own install for use while travelling
- Official releases of the latest ISOs of the different 'flavours' such as Mate, KDE etc
- Special community releases.

It provides the user with the facility to use their own naming convention for each installed OS, so that they know exactly what is presented to them at boot time.

You also have various boot options and can select those you want from a pop-up window. The most popular are:

- **Standard Boot -** which is a straightforward boot of the ISO
- **Boot with Persistence -** which saves changes made while running. Those changes are used when next Persistence Boot is used.
- **Copy to RAM -** which copies the OS into RAM so the boot device is not required and can be withdrawn once the OS is fully booted. This has the effect of slowing the boot as it takes time to copy everything to RAM, but once booted it works much faster as all software is in RAM and not on a slow media stick.

There are some other boot options available also but I would point out that if you install multiple ISOs then your boot list will become very long indeed particularly *if you choose multiple options for each ISO*.

Using MyLiveUSB2 Utility

The utility must be run as root user, so when first launched you need to enter the root password to continue. You are then warned that the device you intend to use should not be plugged into the PC. Follow the instructions carefully, inserting the device when required.

Following the procedure exactly should ensure that the utility operates on the correct device and no other.

From there you meet a variety of pop-up windows which require you to answer questions regarding how and what gets installed.

One item that has caused some confusion is the selection of the '**Name**' by which the installation is displayed in the Grub boot options. This requires careful attention and the steps outlined below must be followed exactly.

To enter your own 'Name' simply click twice on one of the default entries provided, which makes that entry editable. Enter your preferred 'Name'. When you are finished typing in your chosen name, you ***must*** hit the **Enter** key to have your edit accepted in the window.

Your selection must be highlighted prior to clicking OK, to have it applied to the install.

Grub2 is installed only when required .... during the *first* installation of a Live OS. It will install in Legacy and EFI modes as they are available (depends on availability of required partitions).

For additional Live OSs there is no requirement to install Grub2, thus saving wear on a USB flash device.

There is now a **'force Grub2 install'** option near the end of the process.

This has a particular use .... for example ...

If a user creates an MBR device with no ESP partition, just the ext4 partition, then Grub2 gets installed in legacy mode only, when the initial OS is installed.

Should that user now wish to upgrade their device to allow it to boot in EFI mode as well as Legacy, they add an ESP partition to the device (can be placed anywhere – beginning or end) but Grub2 needs to be installed to it.

When a second or subsequent OS is added to the device, simply enabling the 'force Grub2' option, will ensure Grub2 gets installed to the new ESP partition also, and thus provide EFI booting as well as Legacy booting for all the installed OSs.

Selection of ISO or Files for the installation

The utility allows you to select a PCLOS ISO *or* the OS files that would normally be encased in the ISO. You might wish to use Files if you are running the utility from a Live session, or if you have a LiveDVD/CD with the files available but no actual ISO. Where possible an ISO is preferable.

Having selected ISO or Files you are presented with a file-manager window to navigate to and select either *.iso or else a *.sqfs file. Once that is done the utility will continue.

The utility displays a progress bar as it works. Please be aware that the time taken to complete the process is heavily dependent on several factors, such as the write speed of the target device and the size of the ISO or Squashfs file.

While work continues in the background you will get a request for your preferred boot options. These are the options presented to the user by Grub2, when booting the USB device.

You will be informed when the utility has completed all operations, so that you can withdraw the USB device. It should now be bootable in MBR and UEFI modes.

Using the LiveUSB Device

PCs which have ‘MBR compatibility’ enabled, but which default to UEFI booting, will likely display both boot options in their boot list. The UEFI boot mode generally has a UEFI indication in the name of the option entry.

You should now have a USB device with a PCLOS operating system on it ready for live booting.

To add the next or further versions to the USB device just run the utility again, this time ensuring you select to ‘ADD’ to the device when asked, and not to create an initial install.

***

Have Fun with PCLinuxOS