```
$ ls -l
total 60
drwxr-xr-x 2 user user  4096 Apr 12
drwxr-xr-x 2 user user  4096 Apr 12
-rw-r--r-- 1 user user 10000 Apr 12
-rw-r--r-- 1 user user 10000 Apr 12
-rw-r--r-- 1 user user 10000 Apr 12
-rw-r--r-- 1 user user 10000 Apr 12
-rw-r--r-- 1 user user 10000 Apr 12
-rw-r--r-- 1 user user 10000 Apr 12
-rw-r--r-- 1 user user 10000 Apr 12
-rw-r--r-- 1 user user 10000 Apr 12
```

Distributio
Open source
Gnome, KDE,
Command lin
Shell, bash
zsh, Proces
Files, Dire

Linux

The Comprehensive Guide

Michael Kofler

Rheinwerk Computing

Contents

PART II Using Linux

PART IV Text and Code Editors

PART V System Configuration and Administration

14 Basic Configuration 391

18 File System Administration 591

Chapter 3
Installation Instructions

The previous chapter covered the basics of a Linux installation in detail, and now this chapter describes concrete examples. You'll get to know some selected Linux distributions in more detail and learn which special features need to be considered when installing them. At the same time, this chapter provides tips for the first steps after the installation.

The following distributions are described:

- Debian
- Fedora
- Linux Mint
- Manjaro
- openSUSE
- Pop!_OS
- Ubuntu

Server Distributions

Instructions for server installations will follow in Chapter 22, where I describe the installation of AlmaLinux, Debian Server, Oracle Linux, RHEL, Rocky Linux, and Ubuntu Server and go into the details of a server-specific installation, such as RAID and LVM configurations.

Recommendations for desktop use

For Linux beginners, I recommend Ubuntu. This is probably the widest distribution among home users, and in case of problems you can consult numerous forums and wikis.

A good alternative is Linux Mint. This distribution is considered even more beginner-friendly than Ubuntu. Mint has the additional advantage that it doesn't use the Ubuntu-specific Snap packages.

Fedora also rarely causes problems in desktop use, but it's explicitly intended for advanced Linux users. Red Hat views Fedora as an experimental platform. In this respect, Fedora is ideally suited for getting to know the latest developments from the Linux world. The biggest disadvantage of Fedora is its short update period. After one year at the latest, you must perform a distribution update or a new installation.

I used to mention openSUSE at this point, but the currently available version 15.5 is based on SUSE Enterprise 15 and contains some very old software components. The most attractive openSUSE version in my view is the rolling release variant, Tumbleweed.

Many years of updates

An important criterion for the choice of a distribution is its maintenance period: if you want to use Linux without any problems for a number of years, you should choose an LTS version of Ubuntu (20.04, 22.04, etc.) or a RHEL clone (e.g., AlmaLinux). These distributions each offer an update guarantee for at least five years.

This is perfect if you want to install Linux on the computers of your friends and relatives! With some restrictions, the update guarantee also applies to some Ubuntu-derived distributions like Linux Mint or Pop!_OS LTS.

3.1 Debian

Debian: "pure" Linux

No distribution represents "pure" Linux as much as Debian—and that's for several reasons:

- Debian is developed exclusively by a free community of developers. Behind Debian there is neither a company nor commercial interests, but, according to Wikipedia, more than 1,000 developers, most of them working for Debian on a voluntary basis. As a logical consequence, both Debian itself and access to its updates are completely free.
- One of Debian's central goals is that the distribution remains truly "free" in the sense of the open-source idea. The integration of binary drivers or commercial software without freely available source code is taboo. The only exception is firmware files for hardware devices, which have been included by default since version 12, although there is no open-source code for them.
- With Debian, stability takes precedence over very recent versions. That is why an ordinary Debian installation always lags a little behind the current state of development for many important components (kernel, GNOME, server services, etc.). If you need more recent versions, you can install them from the *testing* or *unstable* package sources.
- Debian supports many more hardware platforms than any other distribution. This is another reason why the development of a new Debian version often takes longer than planned.
- The Debian Project is governed by a democratic organization whose leadership members are elected on a regular basis. The rules of the game are formulated in a "social contract" (see *https://www.debian.org/social_contract.1.0.en.html*). This social contract contains the Debian Free

Software Guidelines (DFSG). The guidelines formulate the requirements a software project must meet in order to become part of the official Debian packages.

Debian's importance extends far beyond what can be measured in market share: Debian is an important and indispensable foundation for numerous other distributions, first and foremost for Ubuntu. Many Debian tools, starting with package management, have found their way into other distributions.

Desktop and server use

Debian cannot be clearly classified as a desktop or server distribution; thanks to its universal orientation, Debian is suitable for both purposes. I will go into more detail about some of the special features of server installation in Chapter 22, Section 22.4.

Code names

Debian uses a code name for each version, and each name matches that of a character from the *Toy Story* movie (see Table 3.1). In this book, references to Debian are to Debian 12 unless I explicitly give a different version number.

Code Name	Version	Completion
Buster	Debian 10	July 2019
Bullseye	Debian 11	August 2021
Bookworm	Debian 12	June 2023
Trixie	Debian 13	Expected 2025

Table 3.1 Debian Versions

Versions/variants

Compared to other distributions, Debian doesn't have countless distribution variants. There is only *one* Debian, which consists of a pool of about 64,000 packages. The exact number varies depending on the CPU architecture.

The basic system can be installed either from a DVD image or from a network installation image (`netinst image`, around 350 MiB). This image contains only the installation program. All packages are downloaded from the internet or from a local server during installation.

The hardware support is impressive. While other distributions usually support only two or three CPU platforms, Debian Buster supports 10: besides standard PCs (`amd64` and `i386`), these are ARM (`armel`, `armhf`, and `arm64`), MIPS (`mips`, `mipsel`, and `mips64el`), PowerPC (`ppc64el`), and S390. For some platforms, there are not only installation images, but also live images.

Additional information

Comprehensive information about Debian can be found on the official website:

- *https://www.debian.org*
- *https://www.debian.org/releases/bookworm/amd64*
- *https://www.debian.org/releases/bookworm/amd64/release-notes*

3.1.1 Installing Debian

With Debian, you can use two fundamentally different installation variants: On the one hand, you can write a live image to a USB flash drive and run the very user-friendly installation program included on it. On the other hand, you can use the traditional installation media, which includes the tried and tested installation program. It supports more configuration variants but is much more cumbersome to use and unsuitable for beginners.

Firmware files

Up to and including Debian 11, the "common" images lacked firmware files, which are not available as open-source code. For a modern notebook, this can result in your computer not even being able to establish a network connection. Fortunately, there were modified images *with* the firmware files for such cases.

With version 12, Debian has fortunately opted for a more pragmatic approach and now generally includes the firmware files. This eliminates the confusing differentiation between images that contained only open-source software and images that were simply required for many computers.

3.1.2 Live-Image Installation

Debian provides live images for various desktop systems to choose from (GNOME, KDE, Xfce, etc.). These variants are important because, unlike the conventional installer, they do not allow you to influence the scope of the installation. Thus, the desktop system that runs in the live system is always installed (see *https://cdimage.debian.org/debian-cd/current-live/amd64/iso-hybrid*).

As usual, start the installation by restarting your computer and plugging in the USB flash drive onto which you previously transferred the image. After a few seconds, you will enter the desktop of the live system, where you can first select the keyboard layout and then run **Activities • Install Debian**. When starting the installer, you will be asked for a password. It is `live`. After the start of the installation program, you can select a different language.

Things get exciting in the fourth step, where you decide on the partitioning of your hard disk or SSD (see Figure 3.1). In many cases, the **Install alongside**

option is the right one: if you use this option, the installation program shrinks an existing partition and sets up the partitions required for Debian in the space that's now free.

Figure 3.1 Installation Program Provides Several Partitioning Variants to Choose Among

Partitioning

If necessary, you can opt for **Manual partitioning**. This will give you the option in the next step to set up the desired partitions in the free area of the hard disk or SSD itself, select the file system type you want, and specify the directory (/ for the root partition, `/boot/efi` for the EFI partition, etc.).

Linux account

In the next step, you enter your name and the desired password for your Debian account. Deviating from the usual Debian customs, this user is given `sudo` rights; that is, it is allowed to perform administration tasks later on.

Summary

Before the actual start of the installation, the program displays a summary of all settings that have been made (see Figure 3.2).

Calamares Installer

The *Calamares installer* on the live images is not a Debian in-house development, but comes from a cross-distribution project (see *https://calamares.io/about*).

The installer can be adapted relatively easily to the requirements of the respective distribution. It is also used by various Ubuntu derivatives, by Linux Mint, by Manjaro, and by KDE Neon.

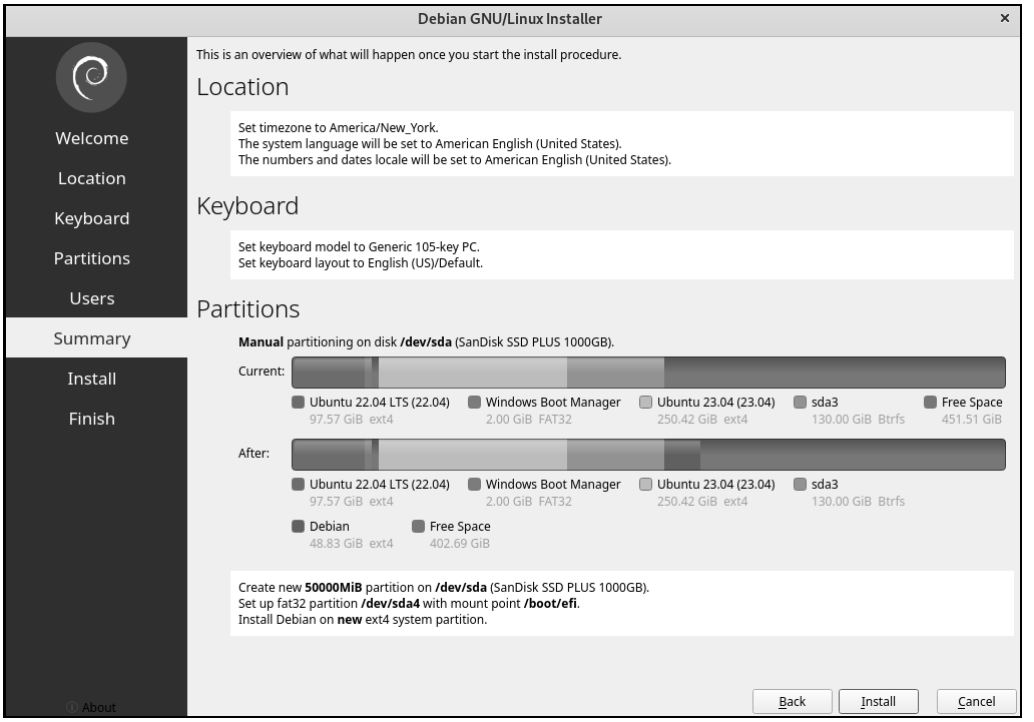


Figure 3.2 Summary before Starting Actual Installation

3.1.3 Standard Installation

Advanced Linux users who want to control the details of partitioning or the scope of installation more precisely are better off using the conventional installation method. You can download suitable images from the following website: *https://debian.org/CD*.

A menu is displayed at the beginning of the installation. By default, the installation program starts in graphics mode. If hardware problems occur, press **Install** to start the installation in text mode or run **Advanced Options • Expert Install**. Then you can have a very precise influence on the individual installation steps and, in particular, on the loading of kernel modules.

root password and user

In the first steps, you will then set the language and keyboard layout. In the dialogs that follow, you need to enter the password for `root` and create a new user.

Automatic sudo Configuration if root Password Is Empty

You can leave the `root` password blank and just click **Next**. In this case, the `root` login remains locked. However, the first user will then be given admin rights, as with Ubuntu, and can switch to `root` mode via `sudo`.

Partitioning the hard disk

The installation program provides the following options for partitioning the hard disk, among others:

- **Guided—Use the Largest Continuous Free Space**
  This option only displays if there is free space on the SSD that is not used by partitions. The installation program then sets up the partitions required for Debian in this area.
- **Guided—Use Entire Disk**
  The installer deletes all partitions and then uses the entire hard disk or SSD for the Debian installation. Another dialog will appear later asking whether you want to store all data in one partition, whether you want a separate home partition (this is recommended), or whether separate partitions should also be created for the `/usr`, `/var`, and `/tmp` directories. The latter is rarely useful.
- **Guided—Use Entire Disk and Set Up LVM**
  As the previous option, but with an LVM system that provides greater flexibility for later changes.
- **Guided—Use Entire Disk and Set Up Encrypted LVM**
  As the previous option, but the LVM system is encrypted. The key must be specified for each boot process; that is, this variant is only conditionally suitable for server installations.
- **Manual**
  This item enables you to perform the partitioning process yourself. However, you can also choose one of the other variants and change the installer's suggestions according to your own ideas.

Regardless of which variant you choose, you must explicitly confirm the partitioning plan again, so there is no risk that the installation program will make the partitioning hastily and irrevocably.

When you perform the partitioning manually, the installation program displays a list of all available partitions (see Figure 3.3). Select existing partitions by double-clicking or pressing Enter. Create new partitions by clicking the **Free Space** item at the bottom of the list. You can also shrink existing Windows and Linux partitions to make more room for new Linux partitions.

Unfortunately, the nested dialogs for editing the partitions are confusing and make little use of the capabilities of a graphical user interface. Many texts in the dialogs are to be interpreted as menu commands and when you click them, they lead to further dialogs. For example, clicking the **Use As: Do Not Use partition** line opens a dropdown list where you specify the desired file system type.

Figure 3.3 Partitioning Hard Disk/SSD

With **Done setting up the partition**, you can save the settings of the last edited partition. After that you can edit another partition or finish the partitioning process by clicking the **Finish Partitioning and write changes to disk** entry below the partition list. The installation program displays a summary of the planned changes to the hard disk partitioning and finally executes them after another confirmation.

If the SSD or hard disk was previously unused, the installer will set up a partition table on its own. On EFI machines, Debian opts for the GPT format. In virtual machines without EFI, Debian stores the partition table in the master boot record (MBR). The installation program doesn't offer any choices in this regard.

Configuring the package manager

In the dialogs that follow, the installation program asks if there are additional installation media (DVDs or additional USB flash drives; there usually are not) or if it should use a mirror server. From this server it can then download packages that are not included in the installation image. The mirror server also serves as a source for updates. Answer **Yes** and select a geographically close server in the subsequent step.

After installing some packages, you will be asked if you want your package selection to be reported to a central server so that the Debian developers can determine the most popular packages.

In the next dialog, you first perform a software selection: the **Debian Desktop Environment** package groups for various desktop systems as well as

Web Server, **SSH Server**, and **Standard System Utilities** are available for selection (shown in Figure 2.3 in Chapter 2).

You can even select multiple desktop systems at the same time in this dialog. Then, every time you log in, you can choose which desktop system you want to use. At the same time, however, choosing multiple desktop systems results in an unnecessarily bloated system with a lot of redundant software, such as multiple audio players, imaging programs, and so on. For this reason, multiple selection is not recommended at this point.

3.1.4 Getting Started

sudo versus su

Depending on how you performed the installation, different rules apply to the way you can subsequently perform admin tasks:

- **With sudo**
  If you installed from the live system or did not specify a `root` password during a standard installation, `sudo` was automatically set up. To execute admin commands, you need to switch into the admin mode in a terminal by using `sudo -s` and specifying your password (see also Chapter 10, Section 10.3).
- **Without sudo**
  Otherwise, if you performed a traditional installation and did specify a `root` password, the user created during the installation is not a member of the `sudo` group. To perform admin tasks, you need to run `su -l` in a terminal window and specify the `root` password.

SSH server

The live system installer does not install an SSH server, and the traditional installer does so only if you have enabled the appropriate option and have not overlooked it. However, if necessary, you can quickly install the SSH server:

```
root# apt update
root# apt full-upgrade
root# apt install openssh-server -y
```

Please Insert the Medium with the Name "Debian GNU/Linux"!

If you installed to a virtual machine, the default installer left a line in `/etc/apt/sources.list` that points to the virtual DVD as the installation media. Now, every time you want to install a package (`apt install`), `apt` wants you to reinsert the installation media. This doesn't make much sense as you want to download the latest version of the package from the internet.

Here is a solution to this: run `sudo gnome-text-editor /etc/apt/sources.list` and remove the line that starts with `deb cdrom`.

Enabling contrib and nonfree package sources

Several official package sources exist for Debian. By default, `main` and `non-free firmware` are active. For the vast majority of packages, this is sufficient.

However, if you also want to install packages whose source code is not freely available or which are based on nonfree libraries, you must add `contrib non-free` to the end of each line in the `/etc/apt/sources.list` file. The file should look like the following example. In the following listing, a `deb` line has been spread over two lines using \ to save space:

```
# in /etc/apt/sources.list
deb http://deb.debian.org/debian bookworm   main non-free-firmware
contrib non-free

deb http://security.debian.org/debian-security/ bookworm-security
main non-free-firmware contrib non-free

deb http://deb.debian.org/debian bookworm-updates  main non-free-
firmware contrib non-free
```

I have refrained from printing the `deb-src` lines here, which only concern the source code packages and are usually not needed.

MP3 and multimedia

Debian can already play MP3 files and the most common audio and video formats after a basic installation. Unofficial packages with additional codecs, multimedia libraries, and programs can be found in Debian-independent package sources—for example, at *https://deb-multimedia.org*.

NVIDIA graphics drivers

The binary drivers for NVIDIA graphics cards are also available as `non-free` packages. Prior to the installation, you must add the `contrib` and `non-free` package sources mentioned earlier to `/etc/apt/sources.list`:

```
root# apt install nvidia-driver
```

For more tips on installing NVIDIA drivers, see Chapter 17, Section 17.3, and the following web page: *https://wiki.debian.org/NvidiaGraphicsDrivers*.

3.2 Fedora

Fedora is a variant of RHEL. Fedora development is supported by Red Hat in terms of personnel and funding. Unlike RHEL, both Fedora itself and all updates are available free of charge. For Red Hat, Fedora is a kind of testing platform to develop and test new features. For many Linux freaks, on the other hand, Fedora is the most modern Linux distribution available. New Linux concepts and ideas are often first found in Fedora before other distributions follow suit. Usually, Fedora is also the Linux distribution that lets you try out the latest GNOME version first.

Despite the development team's willingness to experiment, Fedora has proven to be a relatively stable distribution in recent years. This is obviously where the expertise of the Red Hat developers comes into play. In terms of usability, Fedora has made great progress in recent years: while Fedora used to have the reputation of being "by freaks, for freaks," the distribution is now almost as easy to use as Ubuntu.

The biggest disadvantage of Fedora is its short lifespan: Fedora updates are maintained for the cycle of two releases plus one month. For example, the update period for Fedora 38 ends one month after Fedora 40 is completed. Because the release cycle is normally six months, this corresponds to an update period of only 13 months.

Variants

Fedora is basically available in the three stable variants, *Workstation*, *Server*, and *IoT*. Two more variants are in development:

- *CoreOS* (*https://getfedora.org/en/coreos*) is optimized for use in container environments.
- *Silverblue* (*https://silverblue.fedoraproject.org*) doesn't use the traditional package system; it uses `rpm-ostree` instead. This new package system allows atomic updates as well as undo functions for updates and package installations. It is still open for debate whether Silverblue is the future of Linux or a dead end.

Spins

I'll focus on the workstation variant here, which normally uses GNOME as the desktop. If you prefer another desktop system, you can switch to Fedora Spins. These are Fedora variants with a predefined package selection for alternative desktop systems like KDE, Xfce, LXQt, MATE, Cinnamon, LXDE, or i3 (see *https://spins.fedoraproject.org*).

CPU architectures

Furthermore, there are Fedora variants for other CPU architectures (ARM, Power, and s390x) as well as images prepared for various virtualization and cloud systems (see *https://alt.fedoraproject.org*).

In the near future, there should be a variant of Fedora for Apple computers with the M1, M2, and so on CPUs.

Documentation

For more information about Fedora, you should visit the following websites:

- *https://fedoraproject.org*
- *https://fedoraforum.org*
- *https://docs.fedoraproject.org*
- *https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux*

3.2.1 Installing Fedora

Installation media

At *https://getfedora.org*, you first choose **Fedora Workstation** and then download Fedora Media Writer for Windows, macOS, or Linux. This small program then downloads Fedora and transfers the image to a USB flash drive. Alternatively, you also can find direct download links for the live ISO image at *https://getfedora.org*, which you can then write to a USB flash drive yourself or use as installation media for a virtual machine.

When booting from the installation media, it is scanned for errors by default. You can avoid this time-consuming test by selecting **Start Fedora Workstation** using the cursor keys.

If you install Fedora on a computer with a new NVIDIA graphics card, the installation may get stuck when the graphics mode is activated. Usually, the installation succeeds if you select **Troubleshooting • Start Fedora in Basic Graphics Mode** from the boot menu. You can install the NVIDIA drivers manually at a later time.

In the live system, you can try out Fedora or start the installation program. After setting the language, this combines all configuration settings in a single dialog (see Figure 3.4). Usually, you only need to change a single item—namely, **Installation Destination**.

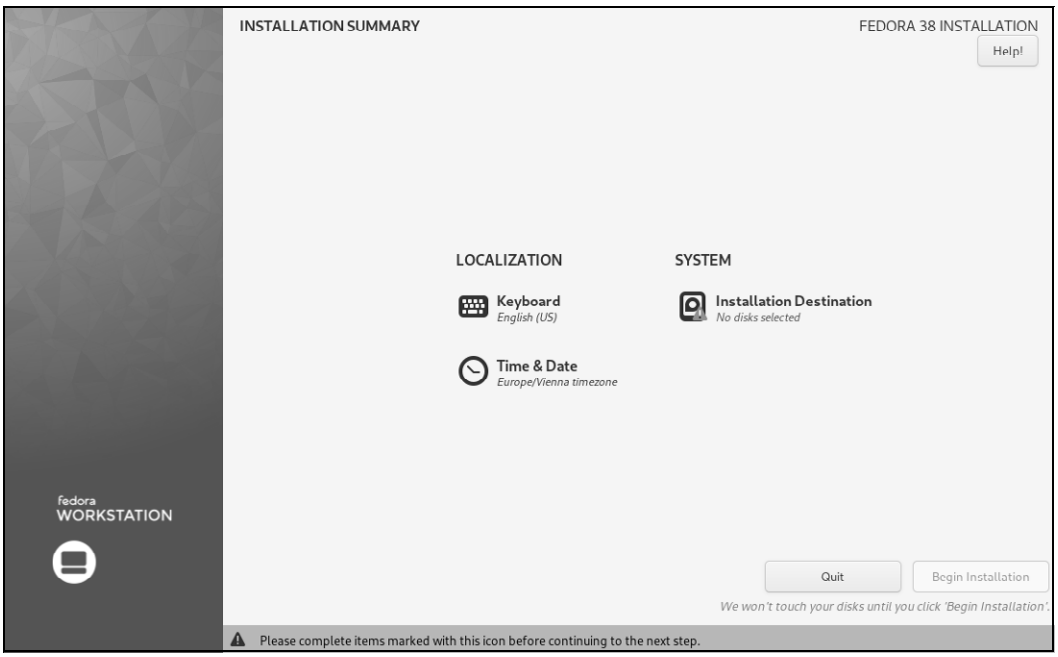


Figure 3.4 Overview of Installation Settings

Partitioning

Clicking the **Installation Destination** icon takes you to a partition editor, which unfortunately is not a masterpiece in terms of intuitive operation. To put it more drastically: I have worked with many partition editors over the last 30 years. The one provided by Fedora/RHEL is the one whose operation is the most illogical.

The first dialog lists the local hard disks and SSDs found (see Figure 3.5). You must make sure that those hard disks on which partitions are to be created or used are marked with a checkmark. (If there is only one disk, then it is already selected, and you only need to click **Done**. The checkmark is decisive. Whether or not the disk has also just been selected with the mouse and is displayed with a blue background, on the other hand, is irrelevant.)

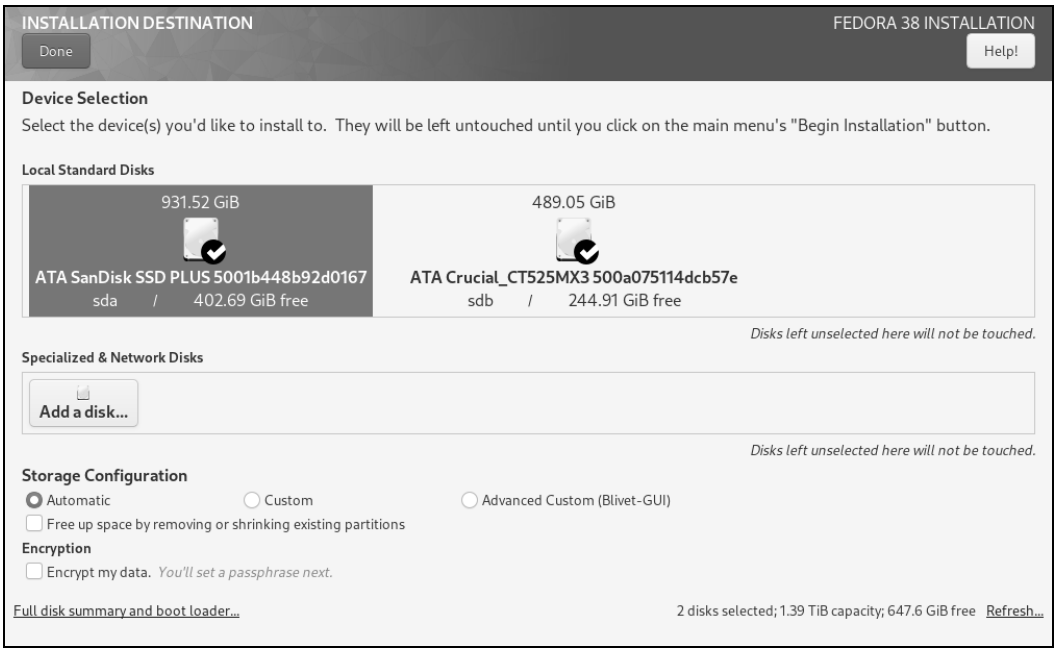


Figure 3.5 Selecting Partitioning Method

Some options enable you to control how you want to proceed:

- If you select **Automatic**, the installation program will take care of suitable partitioning. This works well when you want to make full use of a disk that is still empty—for example, when installing in a virtual machine.
- **Custom** opens an editor in the next step where you can set up the partitions manually and change (or even shrink) existing partitions.
- The **Advanced Custom (Blivet-GUI)** option also takes you to a partition editor designed for advanced users. I personally find this editor easier to

use than the default editor, but possibly that is because I am actually an advanced Linux user. :-)

- The **Free Up Space by Removing or Shrinking Existing Partitions** option is only available in combination with the **Automatic** option. It allows you to delete or shrink some partitions in advance.
- The **Encrypt My Data** option causes the LVM system to be encrypted.

Automatic partitioning

Provided that you have chosen automatic configuration and the hard disk has been empty so far, clicking **Done** actually finishes the configuration and takes you back to the main dialog. Fedora performs the partitioning on its own, but it doesn't find it worth the trouble to tell you the results of its work. Accordingly, this is my task: The installer sets up a 200 to 600 MiB EFI partition, a 1 GiB boot partition, and a third partition that fills the rest of the disk. A file system formatted using `btrfs` ends up in it. For background information on `btrfs` and its Fedora-specific configuration, see Chapter 18, Section 18.11.

Manual partitioning (default program)

There is no way to influence the automatic configuration or to modify it afterward. If you want to perform the partitioning yourself, you must select the **Custom** option. **Done** then leads to another dialog (see Figure 3.6).

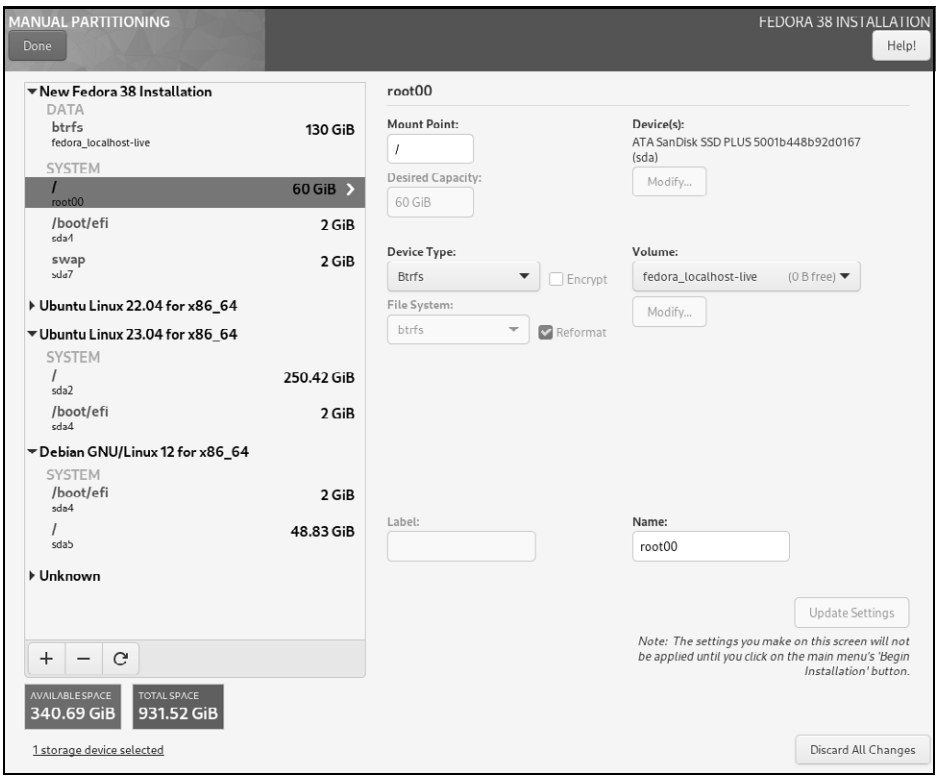


Figure 3.6 Manual Partitioning in Default Editor

The bar on the left lists all partitions or file systems that exist on the disks and those that are to be newly created. The partitions are grouped as follows:

- The first group, which is initially empty, describes the new Fedora system.
- The other groups assign the existing partitions to the already installed operating systems. It may well happen that a partition is displayed in several groups, such as a swap partition that is used by several Linux distributions in parallel.

Figure 3.6 shows an installation on a test machine in parallel with some already existing distributions. A new root partition of 60 GiB and with a `btrfs` file system is supposed to be set up for Fedora. The existing EFI partition, `/dev/sda4`, is to continue to be used under the `/boot/efi` directory. In addition, a new swap partition of 2 GiB is to be created. The remaining existing partitions are not to be addressed under Fedora for the time being.

To create space, you can delete or shrink existing partitions. To set up new partitions, click the plus button (**+**) and specify only two parameters for the time being: under **Mount Point**, enter the mount directory or the name "swap," as well as the required size in MiB or GiB.

Only in the second step do you select the file system type. Under **Device Type**, you need to specify whether the file system is to be set up in an ordinary partition or in a logical volume.

When you have set up all the partitions as you wish, you can finish the process by clicking the **Done** button in the upper-left-hand corner. The installation program then displays a summary of the pending actions. Confirming this data will take you back to the main dialog.

The Hard Disk Will Not Be Changed Until You Have Confirmed

Although the handling of the installation program takes some getting used to, there is at least one advantage: changes you make in the editor are not executed immediately. Rather, the installer waits until you are done with your configuration. Only after a confirmation prompt, which also displays a summary of all outstanding actions, will these be executed.

Manual partitioning (Blivet GUI)

The **Advanced Custom (Blivet-GUI)** and **Done** options take you to an alternative partition editor that I find more intuitive to use. It displays the hard disks, SSDs, and LVM systems found on the computer in the left-hand sidebar (see Figure 3.7).

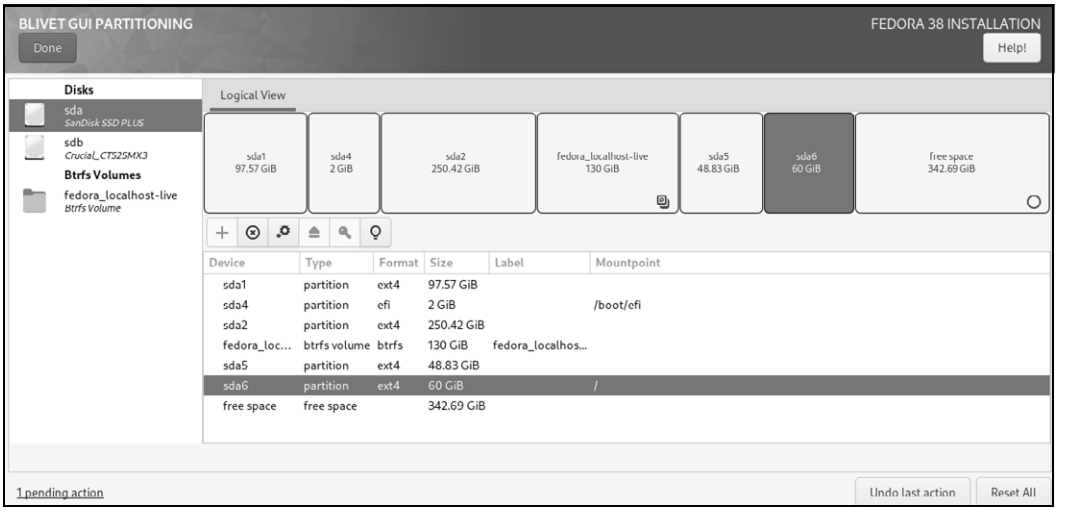


Figure 3.7 Manual Partitioning in Blivet GUI

For the currently selected element, the main area of the editor then displays the list of partitions or logical volumes (LVs). Using buttons and context menu commands, you can then change, set up new, or delete partitions and LVs.

Time zone

The installation program selects a time zone that matches the language setting. If you do not agree with this, you can use **Time & Date** to select another time zone and set the parameters of the NTP server. As soon as you finish the main dialog with **Start Installation**, the actual installation begins.

Configuring users

You must now wait for the actual installation to finish. You only have the option of setting up a user after the first restart. This user will automatically get admin rights and will thus be able to switch to `root` mode via `sudo` (see Chapter 10, Section 10.3). As with macOS and Ubuntu, the `root` user does not receive a password by default and remains locked.

3.2.2 Getting Started

First update

After the first login, you should perform an update—either using the **Software Update** program option or by entering "dnf update" in the terminal. If you choose the first option, the updates will be downloaded first. For their actual installation, the computer will then be restarted.

The first update often takes a similar amount of time as the installation. In this respect, `dnf update` is preferable because then the update is executed during operation. Although some updates (e.g., of the kernel) will only take effect with a reboot in this case, at least you can do other work in parallel.

Enabling the SSH server

By default, an SSH server gets installed, but it does not start automatically. The following command provides an uncomplicated solution to this:

```
root# systemctl enable --now sshd
```

alias settings for root

If you work as `root`, security prompts constantly appear when you run `mv` and `rm` to ask if you really want to perform the operation. These security prompts will stop when you remove the `alias` statements from `/root/.bashrc`.

Setting up additional package sources

In the official Fedora packages, some packages are missing for licensing and patent reasons: drivers for NVIDIA graphics cards, various audio and video codecs, and so on. Fortunately, these packages are available in alternative package sources, the most popular of which is *RPM Fusion* (see *https://rpmfusion.org*).

Setting up those package sources is quite simple: the **Package Sources** menu item of the software program opens a dialog in which you can activate some predefined repositories with one click (see Figure 3.8).

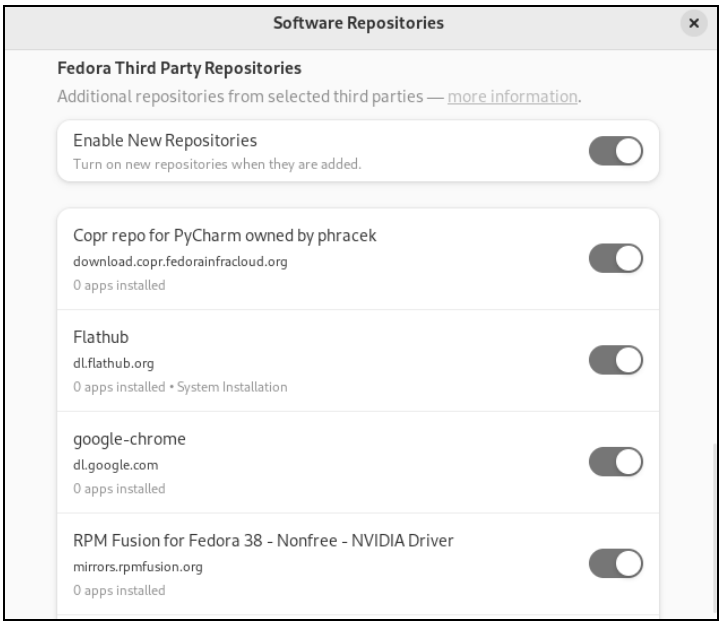


Figure 3.8 Enabling Alternative Package Sources

Multimedia packages

To install various multimedia codecs that are missing in Fedora, you first need to enable the RPM Fusion package sources and then run the following command:

```
user$ sudo dnf install gstreamer*-plugins-bad* gstreamer*-plugins-ugly*
```

NVIDIA drivers

If you depend on the binary graphics drivers from NVIDIA, there are various installation options. As a matter of fact, it should be sufficient to activate the NVIDIA package source in Software and then search for and install the relevant packages in the software as well (shown in Figure 17.3 in Chapter 17). This sometimes worked in my tests, but not always. If necessary, the following command can be executed in the terminal after activating the NVIDIA package source:

```
user$ sudo dnf update -y
user$ sudo dnf install xorg-x11-drv-nvidia akmod-nvidia
```

During the installation, `/etc/X11/xorg.conf` is automatically modified so that the new driver is automatically used after a reboot of the computer. Two alternative ways to install NVIDIA drivers are described in the following web pages:

- *https://negativo17.org/nvidia-driver*
- *https://if-not-true-then-false.com/2015/fedora-nvidia-guide*

Don't be fooled by the year 2015 in the second link! This is when the article was first published. Since then, it has been updated countless times for the current Fedora version.

3.3 Linux Mint

Linux Mint (*https://linuxmint.com*) is a very popular variant for Ubuntu Linux. Linux Mint is based on the same package sources as Ubuntu, generally using only LTS versions of Ubuntu. The Linux Mint version 21.2 tested here is based on Ubuntu 22.04. Instead of the original GNOME desktop, however, Mint uses the MATE or Cinnamon desktop system, depending on the variant.

Linux Mint differs from Ubuntu in other aspects as well:

- Mint uses its own file manager, its own system settings modules, and various smaller add-on programs, and thus it also differentiates itself functionally from Ubuntu. The goal of all in-house developments is simpler operation.
- In addition to the Ubuntu package sources, Mint uses its own package sources. Not only do these serve to provide packages not provided for in Ubuntu, but they also allow Mint to replace individual Ubuntu packages with newer versions.

  Basically, this is a good thing. However, Mint's own package source may prove to be a disadvantage if you set up other package sources (personal

package archives [PPAs], package sources for Docker, etc.) to install external add-on packages. Incompatibilities occur time and again in the process. Especially for software developers, it can therefore be advantageous to stay with the original (i.e., Ubuntu).

- Mint avoids the Snap package format favored by Canonical. For the Firefox web browser, which is now only available as a Snap package on Ubuntu, Mint itself maintains a traditional DEB package.
- The user interface of Mint can be designed much more extensively than that of Ubuntu. Besides icons, miniprograms (called *desklets*) can also be placed on the desktop.

Unlike Ubuntu, Lubuntu, and so on, Linux Mint is not one of the "official" Ubuntu derivatives.

Variants

Linux Mint is available in various flavors, which differ in the desktop system used. (You can choose the proprietary Cinnamon development or MATE or Xfce). In this section, I will only discuss the most popular variant: Linux Mint with the Cinnamon desktop.

Installation

ISO media for Linux Mint is available for download at *https://www.linuxmint.com/download.php*. As with the Debian live system (see Section 3.1), the installation program is based on Calamares and is extremely easy to use.

EFI

Like Ubuntu, Linux Mint uses the `/boot/efi/EFI/ubuntu` directory for installing the boot loader files. This causes one distribution to overwrite the boot files of the other distribution when Ubuntu and Linux Mint are installed in parallel.

Cinnamon

The Cinnamon desktop is basically based on GNOME 3 and its libraries. Instead of `gnome-shell`, Mint runs its own window manager Muffin, and instead of the Nautilus file manager, it runs its fork, Nemo. Visually, Cinnamon looks more like GNOME 2 than GNOME 3, and that is quite intentional. Figure 3.9 shows the Cinnamon desktop of Linux Mint with the characteristic start menu and the welcome wizard, which helps with the installation of programs, proprietary drivers, and multimedia codecs, among other things.

Package management

Package management works like in Ubuntu (see Chapter 16, Section 16.6). However, the `/etc/apt/sources.list.d/official-package-repositories.list` file contains another Mint-specific package source in addition to the Ubuntu LTS package source. At the command level, you can use `apt` to update the entire system as well as install individual packages.

In contrast to Ubuntu, the Snap package system is not available by default, whereas flatpaks are active (see also Chapter 16, Section 16.11).

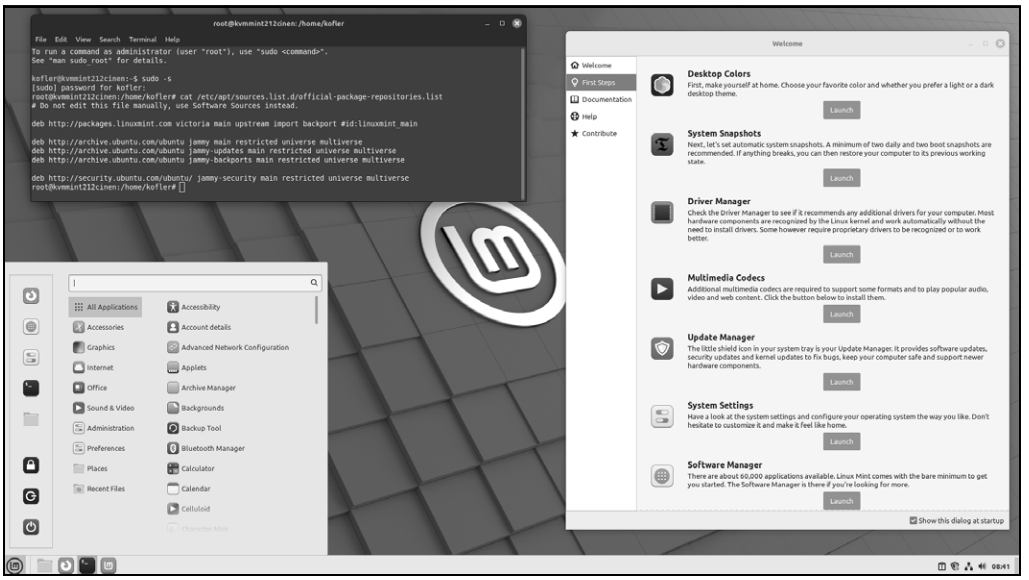


Figure 3.9 Cinnamon Desktop in Linux Mint

The `mintupdate` program takes care of installing updates, and you must confirm each update beforehand. Automation of updates is possible, but the program recommends enabling the Mint-specific *system snapshots* feature upfront: the associated `timeshift` program creates regular backups of the operating system via `rsync`. This function makes it possible to restore a partially destroyed system (e.g., after a package installation) to an older state. However, the snapshot function does not include personal data, so it is not intended for backups.

Mint-specific programs

Besides `mintupdate` and `timeshift`, there are a number of other Mint-specific programs. The easiest way to find their names is to type "mint" in a terminal window and press Tab. The following list names the most important representatives:

- `mintbackup`
  This is a simple backup tool.
- `mintdrivers`
  This supports you with the installation of hardware drivers similar to Ubuntu.
- `mintinstall`
  This helps to install programs. This Mint-specific variant of GNOME's Software takes into account both the Ubuntu and Mint package sources as well as flatpaks.
- `mintstick`
  This helps to write an ISO file to a USB flash drive.

3.4 Manjaro Linux

Arch Linux

Before I can explain the benefits of Manjaro, I must briefly discuss its basic distribution. Arch Linux, which has been available for 20 years, sees itself as a modern rolling release distribution for advanced Linux users. The installation is done in text mode, and a lot of manual work is involved in the configuration. Even Linux professionals still learn a lot in the process. Arch Linux uses its own package management system apart from the established RPM and DEB paths (see Chapter 16, Section 16.7).

Even though I personally find Arch Linux very attractive and have been using it on my most important notebook computer for over two years, the distribution is unsuitable for Linux beginners and is therefore out of place in this chapter. The right time to switch to Arch Linux from another Linux distribution comes after reading Part V of this book. If you have already acquired Linux know-how with other distributions and would like to make friends with Arch Linux, there are many installation guides on the internet (see, for example, *https://wiki.archlinux.org/title/installation_guide*).

Manjaro Linux

There are some distributions derived from Arch Linux. The best known is Manjaro Linux, hereafter *Manjaro* for short. The two main differences compared to Arch Linux are the addition of an easy-to-use installer that is launched from a live system and the integration of a convenient package management and software update program. Nevertheless, most of the advantages of Arch Linux remain. You can download the distribution from *https://manjaro.org*.

The biggest disadvantage of Manjaro is its own package sources. While they protect Manjaro users from mischief caused by updates made too early to Arch packages, they also make manual intervention more difficult. If you need specific additional packages or if problems with the package management occur, you have an advantage with the original (i.e., Arch Linux). The great documentation for Arch Linux doesn't help because it doesn't deal with Manjaro-specific peculiarities.

No UEFI Secure Boot

Neither Manjaro nor the underlying Arch Linux support UEFI Secure Boot. You may need to disable this security feature before starting the installation. Manjaro uses GRUB as a boot loader.

Installation

The installation is performed from within a live system. Depending on which desktop system you want to use, there are several live images for GNOME, KDE, and Xfce to choose from on Manjaro's download page. I tested the GNOME variant for this chapter.

The Manjaro development team already shows its attention to detail in the boot menu of the installation system, where you can set the language,

keyboard layout, and time zone (see Figure 3.10). With most other distributions, the first installation dialogs are always displayed in English until the language setting is done. You also have the option to run the installation system with proprietary drivers, which is especially helpful on computers with modern graphics cards.

Figure 3.10 Language and Time Zone Settings in Boot Menu

In the desktop of the live system, you then start the graphical installation program. Language and keyboard layout are taken from the boot settings, but you have to select the time zone again. You can perform the partitioning either automatically or manually (see Figure 3.11). Unfortunately, because the Manjaro developers love dark colors, the dialogs are quite difficult to read.

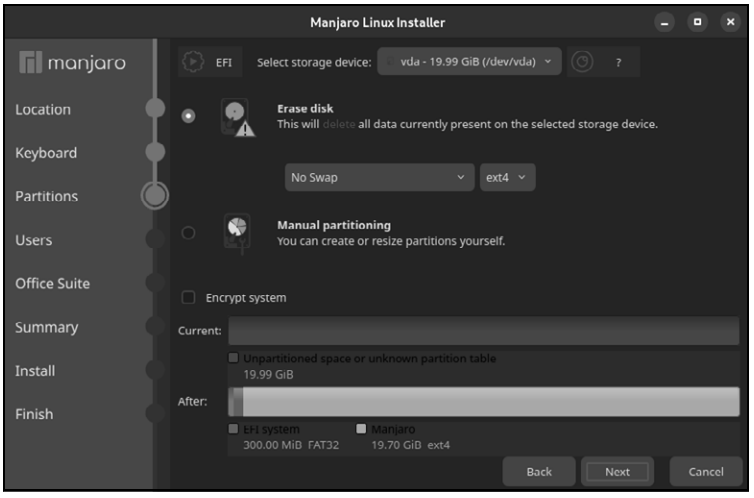


Figure 3.11 Automatic Partitioning of Data Medium

With automatic partitioning, you have the choice of setting up your system without swap space, with a swap partition, or with a swap file, and you choose whether or not you want to encrypt the system partition.

zsh instead of bash

Manjaro is the only one of the distributions presented in this chapter that activates `zsh` by default instead of the otherwise widely used `bash`. Along the way, various shell extensions are also set up; the terminal leaves a somewhat playful impression (as suggested by Figure 8.1 in Chapter 8). The configuration largely corresponds to my personal shell preferences. If you prefer to stick with traditional `bash`, you can simply run `chsh -s /bin/bash` in the terminal and then log in again.

Package management

Like Arch Linux, Manjaro uses the `pacman` command for package management (see also Chapter 16, Section 16.7). Alternatively, however, the *Pamac* graphical software management tool is installed (see Figure 3.12) to help you install and update packages. The website at *https://software.manjaro.org/applications* contains a collection of popular programs that can be installed with the click of a button via Pamac.

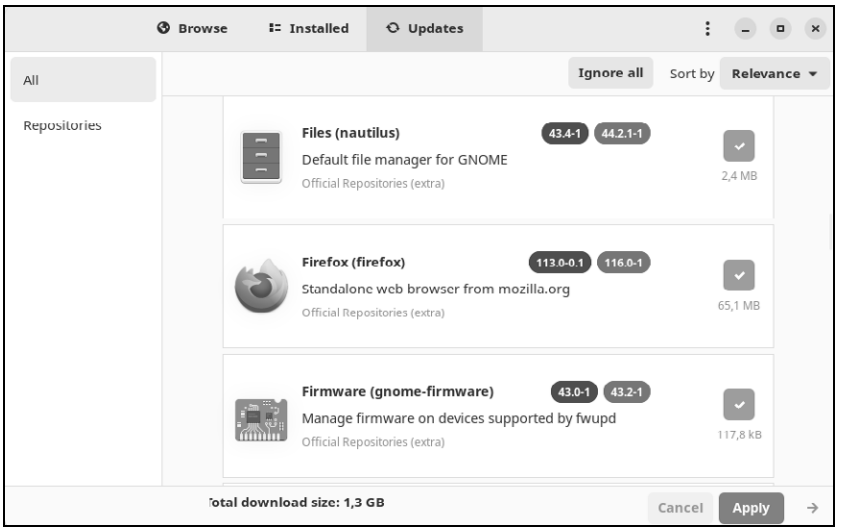


Figure 3.12 Pamac Package Manager

Manjaro uses the same package format as Arch Linux but its own package sources. Arch Linux security updates are applied immediately, other packages often a bit later. The logic as to which packages from Arch Linux are adopted into Manjaro and when is outlined at *https://manjaro.org/features/fresh-and-stable*.

3.5 openSUSE

openSUSE Leap (hereafter mostly shortened to *openSUSE*) is a widely used Linux distribution, especially in German-speaking countries. A key differentiator of the various SUSE distributions compared to their competition is the all-encompassing configuration and administration tool YaST (for *Yet another Setup Tool*).

openSUSE is also considered one of the best KDE distributions (see Figure 3.13). openSUSE thus stands in contrast to most other distributions that primarily focus on GNOME. (Note, however, that the enterprise variants of SUSE are also based on GNOME.)

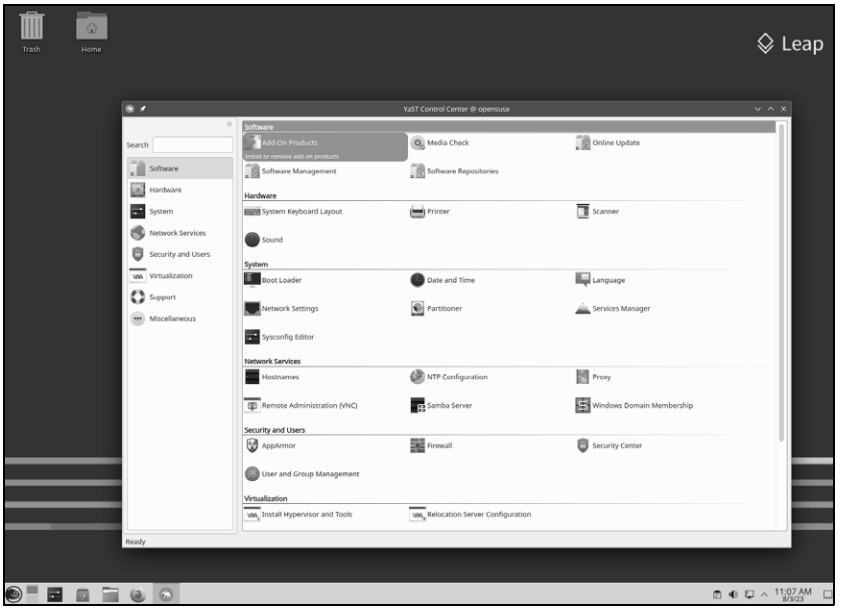


Figure 3.13 KDE Desktop of openSUSE with YaST Configuration Program

The name SUSE

SUSE (then spelled SuSE) was originally a German company, and the abbreviation SUSE originally was derived from *Gesellschaft für Software und Systementwicklung* (Society for Software and System Development). In 2003, Novell acquired SUSE. After that, SUSE changed hands several times before becoming a publicly traded company in 2021.

openSUSE Leap

openSUSE Leap is a free variant of the commercial SUSE distributions. The development is heavily driven by SUSE staff, but there are public beta releases, mailing lists, a bug database, and an active community that participates in and supports development.

SUSE Linux Enterprise Server (SLES) in the same version number serves as the substructure for openSUSE. The kernel, the graphics system, and other

basic components are thus relatively conservatively specified. The openSUSE development team is focused on providing openSUSE with up-to-date versions of programming languages and desktop components (see *https://www.opensuse.org* and *https://doc.opensuse.org*).

Living in the past

In August 2023, when I wrote this chapter, openSUSE 15.5 was the latest version. However, since it is also based on SUSE Linux Enterprise 15 from 2018, some core components are hopelessly outdated. For example, it ships with Python version 3.6, although version 3.11 would be current at that time; `bash` is version 4.4 (currently 5.1); GNOME is version 41 (currently 44); and so on. In this respect, openSUSE 15.*n* appears to be only of limited value at the moment. The rolling release variant Tumbleweed, which I will discuss ahead, is more exciting.

Unclear future

For the next version of Enterprise Linux, SUSE is working on a complete rebuild. SLES 15 is to be replaced by the *Adaptable Linux Platform* (ALP). ALP is based on MicroOS, a new type of *immutable* distribution in which updates take the shape of atomic transactions, not package updates as in the past. However, SUSE ALP is only the core operating system. The applications installed in it should run similarly to containers (think of Docker or Podman).

It is still completely open what this means for the future of openSUSE. First, openSUSE 15.6 is scheduled to be released in 2024 as the last version in the current cycle. To create a new openSUSE version based on ALP, openSUSE is looking for community support. However, because ALP is designed as a pure server platform without desktop packages, this could prove difficult.

Tumbleweed

One very interesting alternative to openSUSE Leap is Tumbleweed. This is an openSUSE variant designed as a *rolling release*: once installed, this distribution constantly receives more current software versions as part of the update system so that (at least in theory) a new installation or a distribution update will never be necessary again.

The Tumbleweed project is based on *Factory*, the development branch of openSUSE. The Tumbleweed developers do make an effort to release new software versions only when the programs run reasonably stably. Nevertheless, when using Tumbleweed, problems are of course to be expected occasionally, as a new software version may still contain bugs or cause incompatibilities with other components.

My experience with Tumbleweed has been quite positive. Tumbleweed contains much more recent software versions (in August 2023: kernel 6.4, bash 5.1, Python 3.11, etc.) than openSUSE Leap.

Note that Tumbleweed does not provide updates to existing packages, but simply new versions of the packages. For this reason, you should use `zypper`

`dup` instead of `zypper up` to update the distribution. For more Tumbleweed tips, visit *https://en.opensuse.org/Portal:Tumbleweed*.

3.5.1 Installing openSUSE

The following installation instructions are for openSUSE Leap 15.5. Basically, however, the instructions also apply to Tumbleweed, whose installation is quite similar. At *https://get.opensuse.org/desktop*, you can find download links for both openSUSE variants and various hardware platforms. You also have a choice between the image with the default installer (the offline image, approximately 4 GiB) and live variants for selected desktop systems. I always use the offline image, which provides more options to influence the installation process.

Starting the installation

In the first dialog of the installation program, you set the language and keyboard layout. On the page that follows, you can add additional package sources and finally specify which function you want your computer to perform. The choices include **Desktop with KDE Plasma**, **Desktop with GNOME**, and **Server**. I have decided to use the KDE desktop here.

Partitioning

The installation program then makes a suggestion for partitioning the hard disk (see Figure 3.14). By default, the program sets up an EFI partition, a swap partition, and a large system partition with the `btrfs` file system. Within the `btrfs` file system, openSUSE creates countless subvolumes.

If you agree with the proposal, just click **Next**. Otherwise, there are three other options:

- **Guided Setup** first opens a dialog where you can enable LVM and disk encryption. In the subsequent dialog, you need to specify whether you want a separate home partition and which file system type you want to use for the root and home file systems.

  The installation program suggests the `btrfs` file system for the root partition, with the **Enable Snapshots** option by default. In combination with the Snapper library, you can then undo admin tasks later. In itself, this is a fine thing. However, the snapshots require a lot of space on the hard disk/SSD and add a lot of complexity that even gives Linux professionals a hard time. You can read the details in Chapter 18, Section 18.11.

  The **Propose Separate Swap Volume** option decides whether a separate partition or logical volume should be provided for the swap space. If you disable this option, a swap file will be set up instead.

- **Expert Partitioner • Start with Current Proposal** takes over the current partitioning proposal. You can then resize the partitions, add more partitions, and so on.

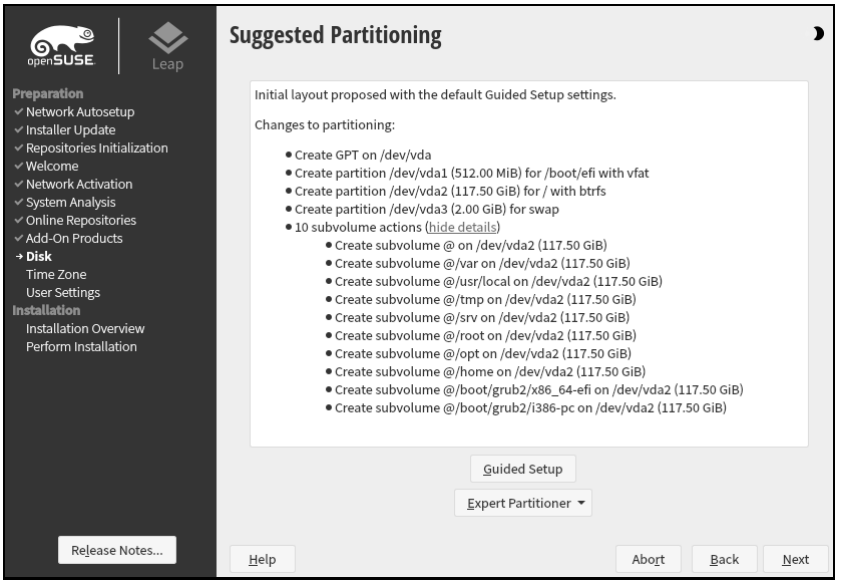


Figure 3.14 Partitioning Suggestion from Installation Program

- **Expert Partitioner • Start with Existing Partitions** leads to the same editor, but now shows only the partitions that already exist on the disks. You can now set up RAID and LVM, create partitions or logical volumes, and so on, to your heart's content (see Figure 3.15).

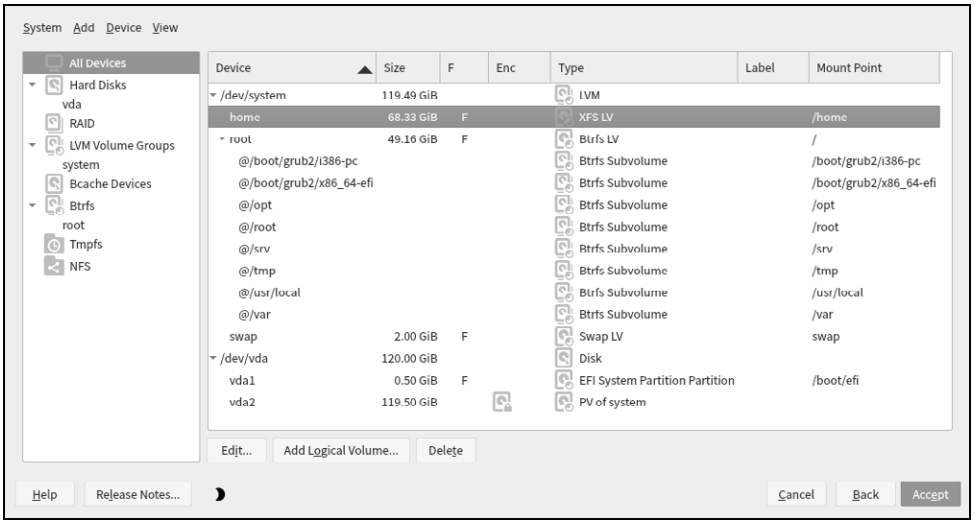


Figure 3.15 Installer's Great Partition Editor

In all three variants, **Back** takes you back to the main dialog where you can choose another option if necessary.

Praise and Criticism

In the SUSE/openSUSE installation program, even complex setups for organizing the file system can be set up quickly and easily. This installation point is organized in a way more user-friendly manner than all other distributions I know. That's great!

However, I don't think it's a good idea to default to `btrfs` with countless subvolumes as the file system type. If you are not (yet) a Linux expert, I strongly advise you to do a **Guided Setup** and set the `ext4` file system type for the root partition!

Basic settings After partitioning, you want to set the time zone and set up a new user. Rather unusual is the fact that the user password is also valid for `root` by default. Only if you disable the related option can you specify your own `root` password in the dialog that follows. The SUSE developers justify their approach with the fact that more than 75% of all users use the same password for `root` as for the first user account anyway. That may be, but from a safety point of view, it is still not ideal!

Summary The installation program then displays a summary of all settings (see Figure 3.16).

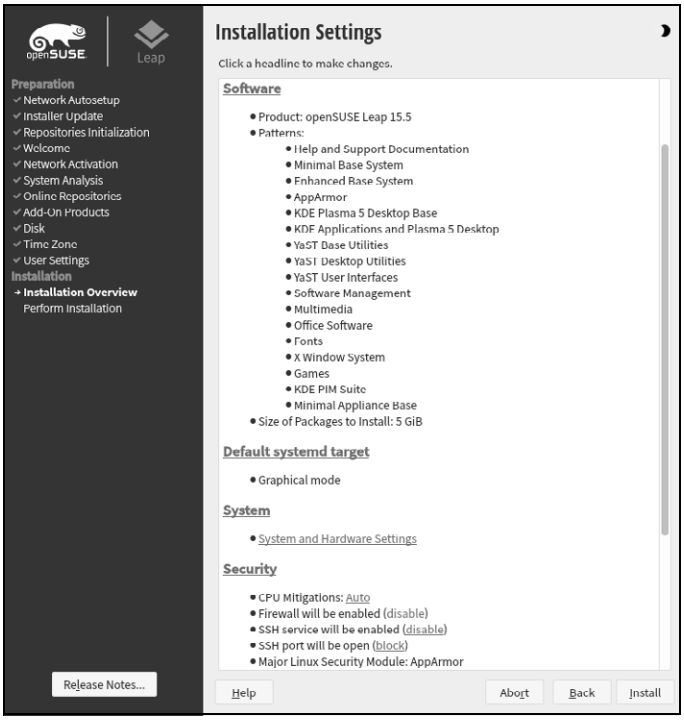


Figure 3.16 Summary of Installation Settings

If you agree, just click **Install** and off you go. However, you should take the trouble to read the installation suggestion at your leisure beforehand! It is often useful or necessary to change details. To make a change, simply click the appropriate item in the summary.

Performing the installation

Once you agree with all the settings, click the **Install** button. The installation takes a few minutes. After that, the computer will be restarted.

3.5.2 Getting Started

Setting the hostname

During the installation, you have no option to set the hostname. openSUSE uses a random string like `linux-q2uf` instead. As a workaround, in the YaST configuration program, open the **System • Network Settings** module. In the **Hostname/DNS** dialog sheet, you now specify the desired hostname. In order for all KDE and GNOME programs to track this change, you must log out and log in again.

Setting the firewall

In openSUSE, a firewall is installed and also active by default. This blocks almost all outgoing network traffic. Among other things, this makes it impossible to access Windows or Samba network shares. The configuration is done by the unfortunately largely useless YaST module at **Security • Firewall**. For this reason, it's better to use the `firewall-cmd` command to set the firewall (see Chapter 29, Section 29.4).

Multimedia

Although openSUSE comes with various audio and video players out of the box, multimedia support is relatively poor as many audio and video formats cannot be played.

The solution to this problem is the Packman package source. To enable it, start the YaST **Software Repositories** module, run **Add • Community Repositories** there, and enable the Packman package source. When doing so, you must confirm the Packman package source key as trusted.

After that, start the YaST **Software Management** module, select the Packman package source via **View • Repositories**, and then click the **Switch System Packages to the Versions in This Repository** link.

NVIDIA graphics drivers

If your computer contains an NVIDIA graphics card, the `nouveau` driver is used by default. The driver works well with many models, but it cannot make optimal use of the graphics card's energy-saving features.

In previous openSUSE versions, installation of the proprietary NVIDIA driver was very easy: to do this, you can use YaST to activate the NVIDIA package source. Next, start the YaST **Online Update** module and then execute the menu command **Extras • Install All Matching Recommended Packages**. YaST then determines which GPU your NVIDIA graphics card contains and installs the appropriate driver package. (There are several different

packages, and it's not very easy to guess which package goes with which graphics card.)

However, during my tests, the activation of the NVIDIA package source failed. An internet search revealed that instead of *https://download.nvidia.com/opensuse/leap/15.5*, the address *http://download.nvidia.com/opensuse/leap/15.5* must be used (i.e., HTTP instead of HTTPS). To set up the package source, you can use **Add • Specify URL**. For more tips on using the NVIDIA drivers on openSUSE, visit *https://en.opensuse.org/SDB:NVIDIA_drivers*.

3.6 Pop!_OS

The Pop!_OS Linux distribution is derived from Ubuntu and is developed by an American company called *System76*. This company sells notebook computers with Pop!_OS installed by default. Pop!_OS can be downloaded free of charge from the following website and then, of course, installed on any computer: *https://pop.system76.com*.

Pop!_OS differs from the Ubuntu base in a surprising number of details:

- **Cosmic DE**
  Currently (i.e., in Pop!_OS 22.04), Cosmic DE is a heavily modified GNOME desktop with its own icons, theme, shortcuts, and various preinstalled extensions. In future Pop!_OS versions, Cosmic DE should become (more) independent from GNOME.
- **Boot process**
  Note that systemd-boot is used instead of GRUB (see Chapter 19, Section 19.5).
- **Package sources**
  There is a separate DEB package source with numerous own or modified packages compared to Ubuntu, as well as Flatpak instead of Snap.
- **Versions**
  After a while of doing semiannual releases, Pop!_OS is now on a two-year cycle. The current Pop!_OS version 22.04 is based on Ubuntu 22.04 LTS. The next version will probably not be available until spring 2024.

Target audience and future

According to its own definition, Pop!_OS is especially optimized for software developers. In the past, the excellent NVIDIA support was also a unique selling point. Meanwhile, setting up a computer with NVIDIA graphics is no longer witchcraft, even with most other distributions.

As a matter of fact, at the beginning I was skeptical about whether Pop!_OS could establish itself in the huge market of Linux distributions in the long

run. After all, the relatively small System76 company has to maintain a wealth of its own software and provide the distribution with bug fixes and software updates. In the past, many distributors have failed at this point. However, you need to give System76 credit for the fact that Pop!_OS maintenance has been working well since 2017 now.

Cosmic DE

But that's not all: System76 is dissatisfied with central parts of the GNOME desktop and has decided to develop its own desktop environment under the name *Cosmic DE*. However, it is unclear when this new, GNOME-independent version of Cosmic DE will actually ship. Most recently, System76 has regularly reported on various Cosmic DE components on its company blog, but there has been no test version or even a release plan available (see *https://blog.system76.com*).

No UEFI Secure Boot

Due to the possible integration of proprietary NVIDIA drivers, Pop!_OS does not support UEFI Secure Boot (even if these drivers are not used at all). You may need to disable this security feature before starting the installation.

3.6.1 Installing Pop!_OS

Pop!_OS is available on the System76 website in two variants. These differ in whether the NVIDIA driver is integrated in the image or not. Note that currently, due to licensing issues, no other major Linux distributor provides the proprietary drivers directly; rather, the drivers must be downloaded during or after installation. Pop!_OS has an advantage here in that the NVIDIA drivers are available right from the start—and this is independent of the network connection, which might still cause problems during the installation. Possibly System76, as a notebook manufacturer with good NVIDIA contacts, has a better negotiating position here.

Pop!_OS is installed from a live system, just like Fedora or Ubuntu. System76 has developed its own installation program and decorated it with its own original pictures (see Figure 3.17).

Whether you'll be happy with the installer depends on the framework:

- The installation process is extremely simple when Pop!_OS is allowed to use the entire SSD and does not have to take into account any other operating systems that may be present.
- In all other cases, and especially in the case of a parallel installation to Windows or to another Linux distribution, you must perform the partitioning manually. This gives Linux professionals great flexibility, such as when integrating existing LVM setups including encryption. However, Linux beginners will be hopelessly overwhelmed at this point.

Figure 3.17 Pop!_OS Installation Options

Using the entire disk

In the first case, after selecting the **Clean Install** option and clicking the desired SSD, you are almost done: you only need to provide a name and password for your account. This password is also used by default to encrypt the file system. If you want, you can set a different password here or dispense with encryption altogether. There are no other questions or options.

In my tests, after rebooting, I had to enter the encryption password on a completely black screen that actually gave the impression that Pop!_OS had crashed. Only after the password was entered blindly did the actual boot process begin, and a few seconds later the desktop appeared.

The solution was a complete update and another reboot. After that, the display of the password dialog during the boot process worked properly.

Manual partitioning

In the second case (i.e., with the **Custom** installation variant), a dialog with all partitions of the disk appears in the next step. However, the installation program doesn't give you the option to change the partitioning. Rather, **Modify Partitions** launches the GParted program (see Chapter 18, Section 18.6). You can then set up partitions using this program.

Pop!_OS requires at least two partitions: the EFI system partition (`/boot/efi`), which must be at least 1 GiB in size and contain a fat32 file system, and a system partition or corresponding logical volume.

If you also want to use LVM or encrypt a partition, you must do this work yourself in the terminal. You will learn the commands required for this in Chapter 18.

You may also have to take care of encryption yourself and run `cryptsetup` in a terminal. However, the process assumes that you are already very familiar with Linux (see Chapter 18, Section 18.19).

Once you've set up all the partitions or logical volumes you want, you need to return to the installation program. It shows all newly created partitions or logical volumes. You can then specify how you want to use the partitions—for example, for the root file system.

The Pop!_OS installation program recognizes the newly created partitions or logical volumes. A few mouse clicks then lead you to configuration dialogs in which you can set how the configured partitions or logical volumes are to be used (see Figure 3.18).

Figure 3.18 Root Partition /dev/vda2 for ext4 File System to Be Set Up

3.6.2 Getting Started

After logging in, the Cosmos desktop shows up as a GNOME system with a large number of modifications and configuration options (see Figure 3.19).

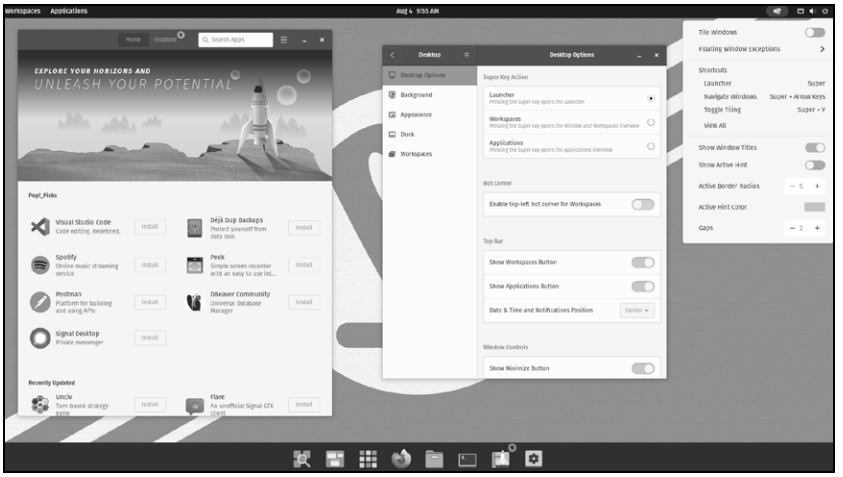


Figure 3.19 Pop Shop for Software Installation (Left); Comprehensive Configuration Options for Desktop (Middle); Menu for Optional Tiling Functions (Right)

Pop!_OS functions

The features Pop!_OS has built into the system menu of the GNOME desktop are noteworthy. On the one hand, you can choose among three different performance levels of the CPU here, and on the other hand, you can switch between Intel and NVIDIA graphics. Behind the scenes, Pop!_OS modifies the boot files for the GPU conversion. Accordingly, the changed GPU setting only takes effect after a reboot, which is a bit tedious in practice. The desired CPU profile or GPU can alternatively be set using the `system76-power` command:

```
user$ sudo system76-power graphics nvidia    (requires a reboot)
user$ sudo system76-power graphics intel     (requires a reboot)
user$ sudo system76-power profile battery
user$ sudo system76-power profile balanced
user$ sudo system76-power profile performance
```

By the way, you can enjoy `system76-power` and its GNOME extension even without Pop!_OS. To do this, set up the System76 package source on Ubuntu and install the `system76` package:

```
user$ sudo sudo apt-add-repository ppa:system76-dev/stable
user$ sudo sudo apt install gnome-shell-extension-system76-power
system76-power
user$ sudo gnome-shell-extension-prefs
```

3.7 Ubuntu

Ubuntu is currently the most popular distribution and the most widely used in the private sector. The motto of Ubuntu is *Linux for human beings*—so in a sense, it's the "human Linux." The Zulu word *ubuntu* also means *humanity toward others* or *mindful cooperation* or *I am what I am because of who we all are*. So Ubuntu is not supposed to be just a lot of software technology, but an entire philosophy.

Canonical

The company behind Ubuntu is Canonical Ltd., which is owned by the South African millionaire Mark Shuttleworth—former owner of Thawte Consulting. Compared to Red Hat, however, Canonical has far fewer employees. In the past, Canonical literally danced on all cylinders and tried to produce Ubuntu versions for smartphones, tablets, TVs, and the cloud in addition to the classic Ubuntu for PCs.

This has been over since 2017. Canonical has since focused on cloud and server customers; that's where Canonical is successful, and that's where

money can be made. The development of a standalone Linux desktop (*Unity*) was discontinued in favor of the GNOME desktop. In general, the budget for optimizations and in-house developments in the desktop area has obviously been severely limited since then.

Standard versions

There are new Ubuntu releases every six months, with the version number reflecting the date of completion. So Ubuntu 23.04 refers to the Ubuntu version completed in April 2023. Also, each Ubuntu version has a strange codename; for version 21.04, for example, it is *Hirsute Hippo*. These code names are perfect for search queries! A search for "hippo nvidia" will return much more specific results than a search for "ubuntu nvidia" or even "linux nvidia".

There is only a nine-month update service for the semiannual Ubuntu versions. Thus, these versions are actually only recommended to Linux professionals who are not bothered by regular distribution updates.

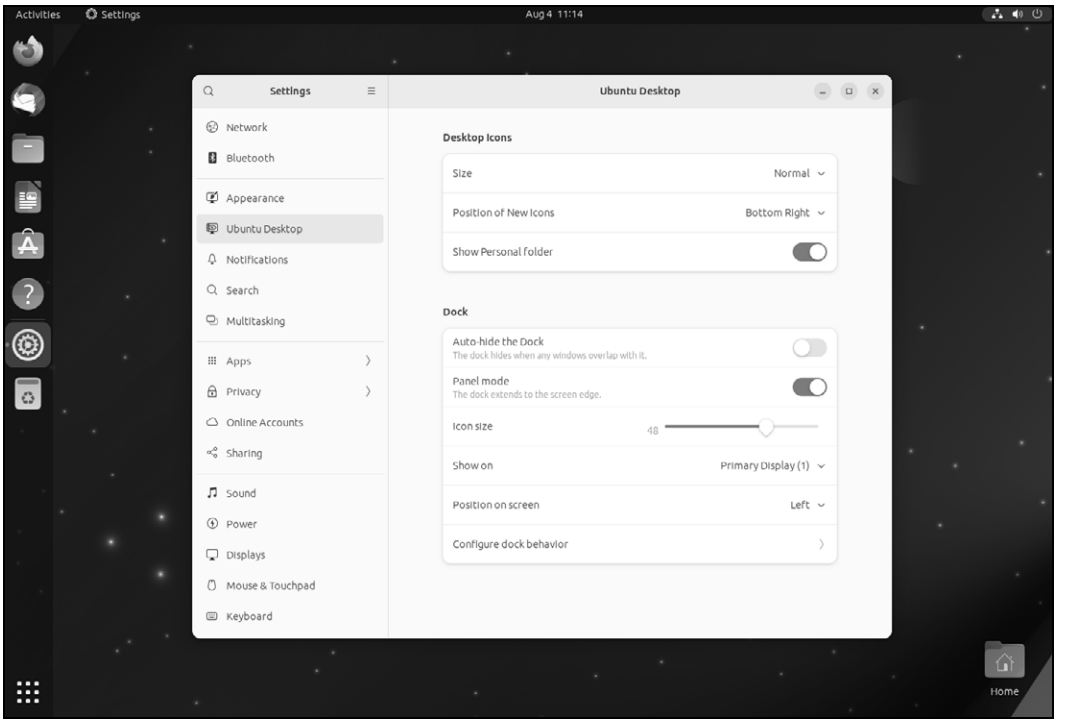


Figure 3.20 Ubuntu's Slightly Modified GNOME Desktop

LTS versions

The long-term support (LTS) versions, whose desktop packages and server packages are maintained for five years, occupy a special position among the

many Ubuntu versions. When I finished this book, 22.04 was the last available LTS version. The next LTS version, Ubuntu 24.04, will not be available until April 2024.

Variants

There are numerous distributions derived from Ubuntu. Table 3.2 summarizes the most important official—that is, supported by Canonical—ones, as well as some unofficial variants.

All official variants use the same package sources and can therefore be extended as you like. You can also install Xubuntu first and add the GNOME packages from Ubuntu later. The main difference between the different Ubuntu variants is which package selection is included on the disk and installed for the first time. The unofficial Ubuntu variants also provide additional package sources with additional packages. They contain programs that are missing from the official package sources or are included there in a different (mostly older) version.

Variant	Description
Kubuntu	Ubuntu with KDE
Ubuntu MATE	Ubuntu with the MATE desktop
Xubuntu	Ubuntu with Xfce
Lubuntu	Ubuntu with LXDE
Ubuntu Server	Ubuntu for server use (see Chapter 22, Section 22.3)
Ubuntu Studio	Ubuntu for multimedia users
Ubuntu Budgie	Ubuntu with Budgie desktop
Linux Mint	Popular Ubuntu variant without Unity (unofficial)
Pop!_OS	Ubuntu variant by System76 with especially good support for notebook computers with NVIDIA graphics (unofficial)
Elementary OS	Ubuntu variant with macOS-like desktop (unofficial)
Zorin OS	Desktop variant, which is specifically aimed at Windows migrants (unofficial)

Table 3.2 Ubuntu Variants

3.7.1 Installing Ubuntu

Installation media

At *https://www.ubuntu.com/download*, you can download free Ubuntu installation media. The ISO images must then be transferred to a USB flash drive.

Alternative Ubuntu Downloads

At *http://cdimage.ubuntu.com*, you will find various alternative installation media. This includes the daily images of the next Ubuntu version and variants currently under development.

Starting the standard installation

To perform a standard Ubuntu installation, you want to reboot the computer with the Ubuntu DVD or an appropriate USB flash drive. If you install on a computer with an NVIDIA GPU, you should select the **Safe Graphics** variant in the boot menu; otherwise, the graphics system may fail to start.

New installer

The installer described in this section has only been in use since version 23.04. In my tests, the installer worked wonderfully in virtual machines, but failed in installations on real hardware or more complex setups. By the time this book is published, there will already be the next Ubuntu version available, in which the worst teething troubles of the installer are presumably fixed.

In the live system, you can now both try out and install Ubuntu. After starting the installation program, you need to select the relevant language in the first step. In the next dialog with the basic settings (see Figure 3.21), the **Minimal Installation** option reduces the installation scope somewhat. After the installation, the complete basic system is available, but application programs like Thunderbird, LibreOffice, and an audio player are missing. These programs can be installed later if needed.

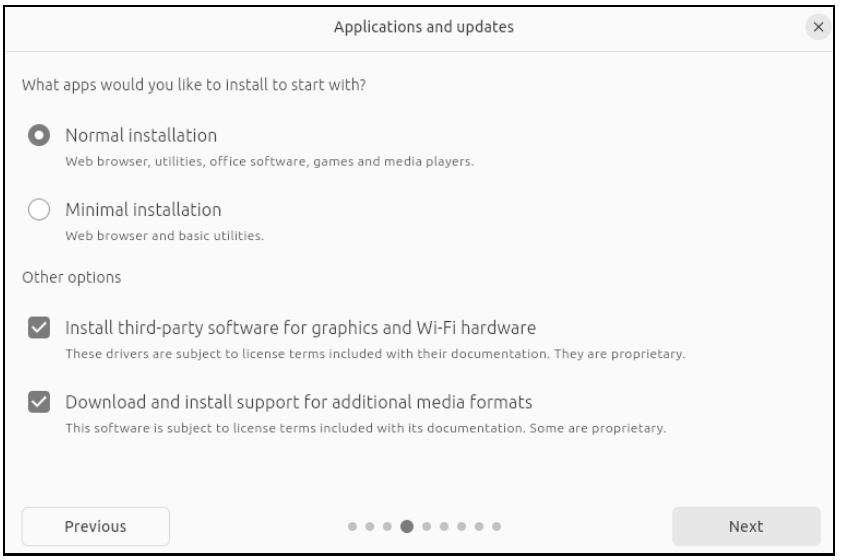


Figure 3.21 Basic Settings in Ubuntu Installer

The **Install Third-Party Software...** option controls whether various drivers that are not available as open-source software should be installed. Among other things, this affects the proprietary NVIDIA graphics drivers. If the installation program detects an NVIDIA GPU, the drivers are installed and are then already available at the first start.

Support for Additional Media Formats adds audio and video codecs to the installation. These codecs are necessary for the playback of video files to work.

Installation type

Depending on which other operating systems the installer detects on the SSD, the program then offers several installation variants to choose from (see Figure 3.22).

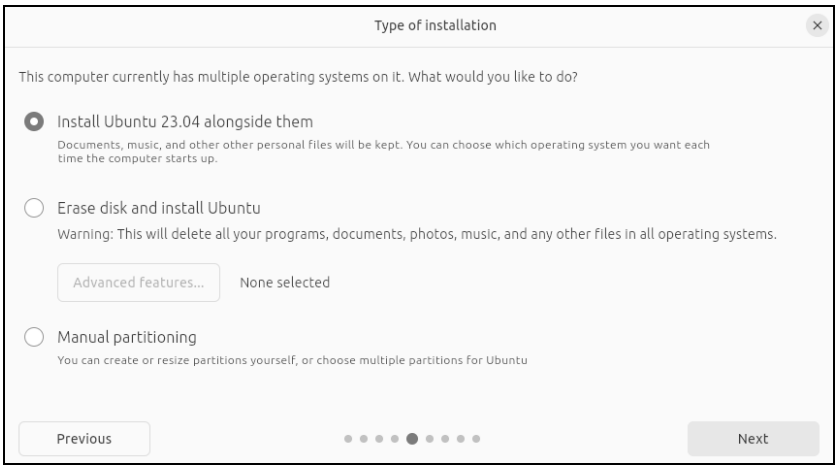


Figure 3.22 Setting Installation Type

The installer always shows only the options that are relevant for the computer at hand.

- **Install Ubuntu alongside Them**
  In this variant, the installer uses free areas on the SSD and installs Ubuntu there. After installation, you can select which operating system to boot from the EFI boot menu each time you reboot.
- **Erase Disk and Install Ubuntu**
  This will erase the entire hard drive or SSD and then repartition it.
- **Manual Partitioning**
  This option allows you to perform the partitioning yourself.

For most options (with the exception of **Manual Partitioning**), the installer creates a new system partition that fills the entire free space of the disk. You cannot influence the partitioning.

Manual partitioning

If you want to set the size of the partitions yourself, want your own `/home` partition, and so on, then you need to select **Manual Partitioning**.

To create a new partition, first select the **Free Space** entry and then click the plus button (**+**). In the dialog that appears, you must specify the type of partition, the size, and the file system (see Figure 3.23).

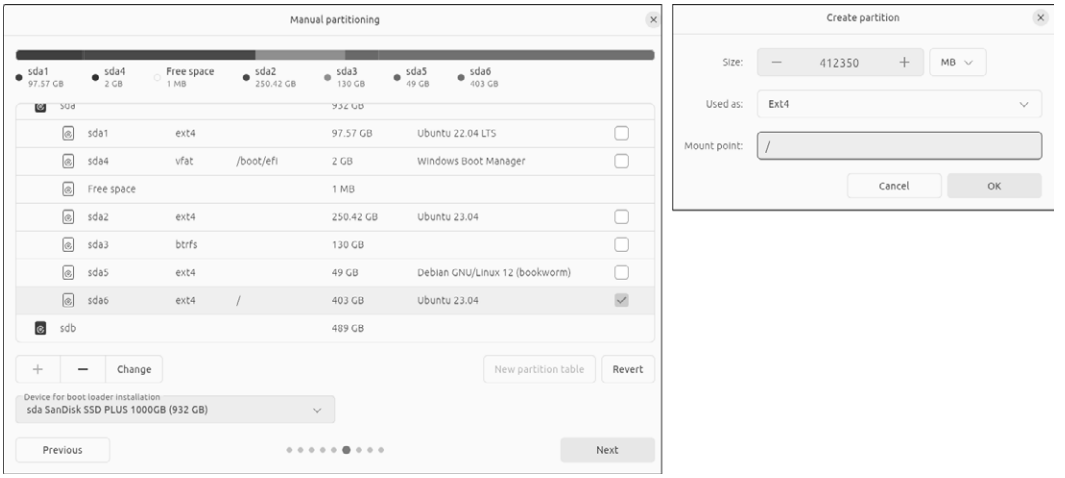


Figure 3.23 Manual Partitioning

If there is already a suitable partition on your hard disk where you want to install Ubuntu, select this partition, change the mount point (the mount directory), and, if necessary, activate the checkbox to reformat the partition.

You don't have to worry about the EFI system partition. As soon as you manually set up the first partition, the program automatically adds an ESP sized at 1 GiB.

After a final query, finish the partitioning by clicking the **Install Now** button. After that, you will not be able to stop the installation or change the partitioning.

Time zone

You can now make various remaining settings parallel to the installation. First, specify the time zone you're in. The installation program assumes that your computer's clock is set to the local time.

Configuring users

In the next step, specify the username and password for the first Ubuntu user. You can set up additional users later during operation if necessary. In contrast to other Linux installers, you do not need to specify a `root` password here: admin work can be done on Ubuntu by an ordinary user via `sudo`.

Desktop theme

In the last dialog, you can choose between a light and a dark appearance for the desktop. If necessary, you can revise this setting later during operation.

3.7.2 Getting Started

Performing updates

To bring all installed Ubuntu packages up to date, run the following command in a terminal window:

```
user$ sudo apt update
user$ sudo apt full-upgrade
```

ubuntu-restricted-extras

If you did not enable **Support for Additional Media Formats** during installation, then installing the `ubuntu-restricted-extras` package will make your Ubuntu system multimedia ready. Among other things, this installs codecs for all possible audio and video formats, including MP3:

```
user$ sudo apt install ubuntu-restricted-extras
```

Installing hardware drivers

Ubuntu provides proprietary hardware drivers for NVIDIA graphics cards as well as for some Wi-Fi adapters. To install them, open the **Applications & Update** program from the **Start** menu. The drivers suitable for your system are listed in the **Additional Drivers** dialog (shown in Figure 17.2 in Chapter 17). To activate, you must restart the computer.

Alternatively, the installation of such drivers can also be done in text mode, which is very convenient, especially when the graphics system doesn't work due to a lack of drivers. In this case, you first need to determine the list of available drivers using `ubuntu-drivers devices` and then install the recommended driver via `apt`:

```
root# ubuntu-drivers devices
vendor   : NVIDIA Corporation
model    : GP107GLM [Quadro P1000 Mobile]
...
driver   : nvidia-driver-470 - distro non-free
driver   : nvidia-driver-525 - distro non-free recommended
driver   : nvidia-driver-535 - distro non-free
driver   : xserver-xorg-video-nouveau - distro free builtin
root# apt install nvidia-driver-525
```

Kernel updates (LTS enablement stack)

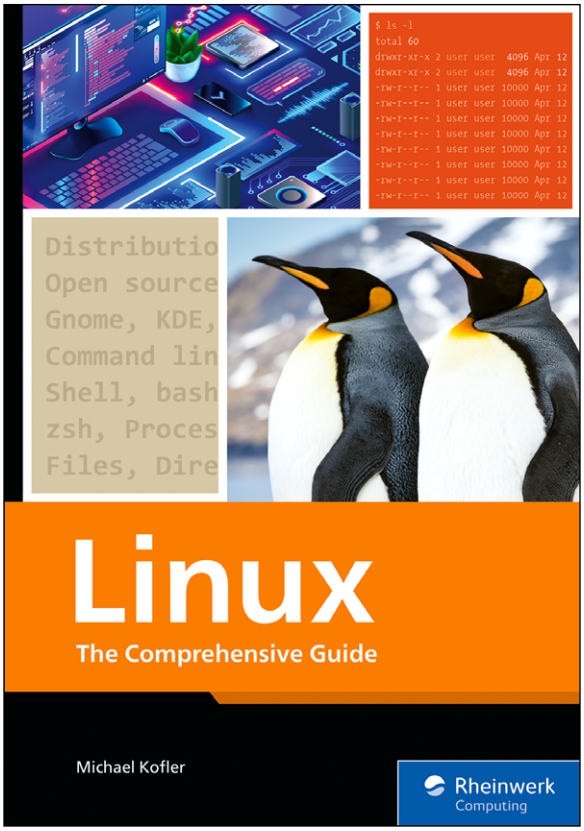
In the course of the usual updates for Ubuntu LTS, you only get bug fixes for the kernel. However, the base version of the kernel remains unchanged by default throughout the lifetime of Ubuntu LTS. Things are a little different when you perform a fresh LTS installation: Canonical updates the installation media (Ubuntu 22.04.1, 22.04.2, etc.) about twice a year, sometimes

using more recent kernel versions. This increases the compatibility of Ubuntu LTS with new hardware.

But how do you get newer kernel versions on an old Ubuntu LTS installation? For this, you must explicitly install hardware enablement (HWE) packages. The following command applies to Ubuntu 22.04 LTS; for Ubuntu 24.04, you must adjust the version number accordingly:

```
root# sudo apt install --install-recommends linux-generic-hwe-22.04
```

Note, however, that this will put you into a rolling release mode for the kernel! When the next Ubuntu LTS update version is released (22.04.3, 22.04.4, etc.), you will again receive the then-current kernel as part of the usual updates!



Michael Kofler

Linux

The Comprehensive Guide

- Your complete, cross-distribution, professional guide to Linux, for beginners and advanced users
- Get detailed instructions for installation, configuration, and administration, on both desktop and server
- Set up security, virtualization, and more

rheinwerk-computing.com/5779

Dr. Michael Kofler is a programmer and Linux administrator. He has been one of the most successful and versatile computing authors in the German-speaking world for many years. His current topics include Linux, Docker, Git, hacking and security, Raspberry Pi, and the programming languages Swift, Java, Python, and Kotlin.

ISBN 978-1-4932-2499-9 • 1178 pages • 05/2024

E-book: $54.99 • **Print book:** $59.99 • **Bundle:** $69.95

Rheinwerk Publishing