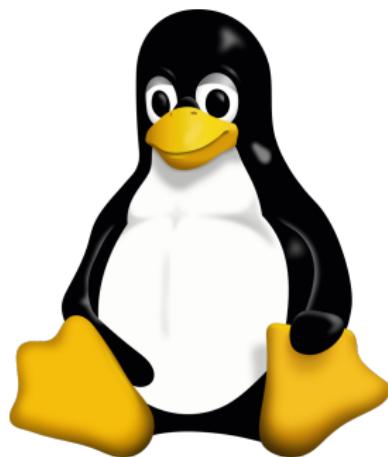


Getting Started with Linux

Zoltán Szabó @ Department of Statistics, LSE (Sept. 26, 2025)



Contents

- Desktop tour ✓
- Applications.
- A bit of Linux history.
- Linux distributions.
- Installation.
- Ricing and phones.

Applications: categorized; some handy ones

Notations:

- link. OD := Odysee, YT := YouTube (as fallback).
- M = 'in main', A = 'in AUR', W = web client, p = proprietary.

Web & mail:

- browser:
 - librewolf-bin[†] (A; ∃ uBlock Origin), tor-browser-bin (A); to keep an eye on: Ladybird!

[†] Librewolf: the best browser? [YT]

Applications: categorized; some handy ones

Notations:

- link. OD := Odysee, YT := YouTube (as fallback).
- M = '∈ main', A = '∈ AUR', W = web client, p = proprietary.

Web & mail:

- browser:
 - librewolf-bin[†] (A; ∃ uBlock Origin), tor-browser-bin (A); to keep an eye on: Ladybird!,
 - Chromium-based browsers: warning! ⇐ Manifest v3 [OD₁, YT₁; YT₂].



The Internet

The Internet
when you
block ads

[†] Librewolf: the best browser? [YT]

Applications: categorized; some handy ones

Notations:

- **link.** OD := Odysee, YT := YouTube (as fallback).
- M = '∈ main', A = '∈ AUR', W = web client, p = proprietary.

Web & mail:

- browser:
 - librewolf-bin[†] (A; ∃ uBlock Origin), tor-browser-bin (A); to keep an eye on: Ladybird!,
 - Chromium-based browsers: warning! ⇐ Manifest v3 [OD₁, YT₁; YT₂].



The Internet

The Internet
when you
block ads

- e-mail: ProtonMail (W), thunderbird (M).

ProtonMail: get a free month on the Mail Plus plan!

[†] Librewolf: the best browser? [YT]

Applications+

Media:

- image:
 - viewer: `feh` ([M](#)), `gthumb` ([M](#)).
 - editor: `gimp` ([M](#); bitmap), `inkscape` ([M](#); vector).

Applications+

Media:

- image:
 - viewer: `feh` (M), `gthumb` (M).
 - editor: `gimp` (M; bitmap), `inkscape` (M; vector).
- video (& audio):
 - player: `mplayer` (M), `vlc` (M), `mpv` (M).
 - private YouTube viewer: `freetube` (A).
 - editor: `kdenlive` (M).
 - recording, live streaming: `obs-studio` (M).
 - downloader: `yt-dlp` (M).

Applications+

Media:

- image:
 - viewer: `feh` (M), `gthumb` (M).
 - editor: `gimp` (M; bitmap), `inkscape` (M; vector).
- video (& audio):
 - player: `mplayer` (M), `vlc` (M), `mpv` (M).
 - private YouTube viewer: `freetube` (A).
 - editor: `kdenlive` (M).
 - recording, live streaming: `obs-studio` (M).
 - downloader: `yt-dlp` (M).
- audio:
 - player: `mpd`, with `mpd-mpris`, `rmpc`, `kitty`, and `playerctl` (all M).
 - editor: `tenacity` (M).



- Text:

- document viewer: `xdvi` ∈ `texlive` (M, group), `xpdf` (M), `okular` (M),
- .pdf annotation: `xournalpp` (M),
- text editing: `texlive` (LATEX, M, this group), `kile` (M) (\approx WinEdt_P), `vim` (M), `libreoffice-still` (M).

Applications+

- Text:
 - document viewer: `xdvi` ∈ `texlive` (M, group), `xpdf` (M), `okular` (M),
 - .pdf annotation: `xournalpp` (M),
 - text editing: `texlive` (LATEX, M, this group), `kile` (M) (≈ `WinEdtp`), `vim` (M), `libreoffice-still` (M).
- Chat & collaboration:
 - chat: `zoomp` (A), `teamsp` (A), `BigBlueButton` (W), `Jitsi Meet` (W), `linphone-desktop-appimage` (A),
 - version control: `git` (M),
 - calendar & reminder: `remind` (M).



Network:

- firewall: `ufw` (**M**).

Network:

- firewall: [ufw](#) (M).
- remote access:
 - VNC client & server: [tigervnc](#) (M),
 - SFTP client: [filezilla](#) (M), [remmina](#) (M; VNC client too).

Network:

- firewall: [ufw](#) (M).
- remote access:
 - VNC client & server: [tigervnc](#) (M),
 - SFTP client: [filezilla](#) (M), [remmina](#) (M; VNC client too).
- VPN: [proton-vpn-gtk-app](#) (M).

Network:

- firewall: [ufw](#) (M).
- remote access:
 - VNC client & server: [tigervnc](#) (M),
 - SFTP client: [filezilla](#) (M), [remmina](#) (M; VNC client too).
- VPN: [proton-vpn-gtk-app](#) (M).
- network statistics collector: [vnstat](#) (M).

Network:

- firewall: [ufw](#) (M).
- remote access:
 - VNC client & server: [tigervnc](#) (M),
 - SFTP client: [filezilla](#) (M), [remmina](#) (M; VNC client too).
- VPN: [proton-vpn-gtk-app](#) (M).
- network statistics collector: [vnstat](#) (M).
- network speed tester: [speedtest-cli](#) (M).

Network:

- firewall: `ufw` (M).
- remote access:
 - VNC client & server: `tigervnc` (M),
 - SFTP client: `filezilla` (M), `remmina` (M; VNC client too).
- VPN: `proton-vpn-gtk-app` (M).
- network statistics collector: `vnstat` (M).
- network speed tester: `speedtest-cli` (M).
- USB-LTE modem handling: `usb_modeswitch` (M).

Network:

- firewall: `ufw` (M).
- remote access:
 - VNC client & server: `tigervnc` (M),
 - SFTP client: `filezilla` (M), `remmina` (M; VNC client too).
- VPN: `proton-vpn-gtk-app` (M).
- network statistics collector: `vnstat` (M).
- network speed tester: `speedtest-cli` (M).
- USB-LTE modem handling: `usb_modeswitch` (M).
- validating, recursive, caching DNS resolver: `unbound` (M).

- Security:
 - password manager: `keepassxc` (**M**).

- Security:
 - password manager: `keepassxc` (M).
- News:
 - RSS/Atom feed reader: `newsboat` (M).

- Security:
 - password manager: `keepassxc` (M).
- News:
 - RSS/Atom feed reader: `newsboat` (M).
- Environment:
 - desktop: `plasma` (M, group), `qtile` (M, WM), `gnome` (M, group).
 - terminal: `tilix` (A).

Applications+

- Security:
 - password manager: `keepassxc` (M).
- News:
 - RSS/Atom feed reader: `newsboat` (M).
- Environment:
 - desktop: `plasma` (M, group), `qtile` (M, WM), `gnome` (M, group).
 - terminal: `tilix` (A).
- File manager: `thunar` (M).

- Security:
 - password manager: `keepassxc` (M).
- News:
 - RSS/Atom feed reader: `newsboat` (M).
- Environment:
 - desktop: `plasma` (M, group), `qtile` (M, WM), `gnome` (M, group).
 - terminal: `tilix` (A).
- File manager: `thunar` (M).
- Archive manager: `engrampa` (M).

- Security:
 - password manager: `keepassxc` (M).
- News:
 - RSS/Atom feed reader: `newsboat` (M).
- Environment:
 - desktop: `plasma` (M, group), `qtile` (M, WM), `gnome` (M, group).
 - terminal: `tilix` (A).
- File manager: `thunar` (M).
- Archive manager: `engrampa` (M).
- Calculator: `gnome-calculator` (M).

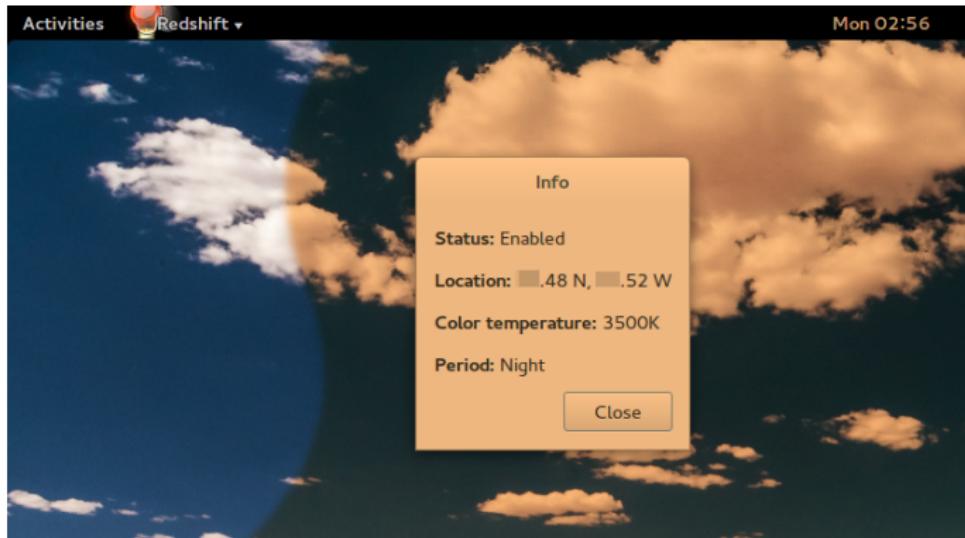
Applications+

- Security:
 - password manager: `keepassxc` (M).
- News:
 - RSS/Atom feed reader: `newsboat` (M).
- Environment:
 - desktop: `plasma` (M, group), `qtile` (M, WM), `gnome` (M, group).
 - terminal: `tilix` (A).
- File manager: `thunar` (M).
- Archive manager: `engrampa` (M).
- Calculator: `gnome-calculator` (M).
- Programming: `python` (M), `jupyter-notebook` (M), `spyder` (M).

Applications+

Utilities:

- eye protection: **redshift** (M),



Utilities:

- eye protection: `redshift` (M),
- notifications:
 - notification daemon: `dunst` (M).
 - xmessage clone: `gxmessage` (M).

Utilities:

- eye protection: `redshift` (M),
- notifications:
 - notification daemon: `dunst` (M).
 - xmessage clone: `gxmmessage` (M).
- disk usage analyzer: `dysk` (M), `baobab` (M).

Utilities:

- eye protection: `redshift` (M),
- notifications:
 - notification daemon: `dunst` (M).
 - xmessage clone: `gxmmessage` (M).
- disk usage analyzer: `dysk` (M), `baobab` (M).
- drive benchmarking: `gnome-disk-utility` (M).

Utilities:

- eye protection: `redshift` (M),
- notifications:
 - notification daemon: `dunst` (M).
 - xmessage clone: `gxmmessage` (M).
- disk usage analyzer: `dysk` (M), `baobab` (M).
- drive benchmarking: `gnome-disk-utility` (M).
- launcher: `dmenu` (M; ∈ suckless tools), `rofi` (M).

Utilities:

- eye protection: `redshift` (M),
- notifications:
 - notification daemon: `dunst` (M).
 - xmessage clone: `gxmmessage` (M).
- disk usage analyzer: `dysk` (M), `baobab` (M).
- drive benchmarking: `gnome-disk-utility` (M).
- launcher: `dmenu` (M; ∈ suckless tools), `rofi` (M).
- screen locker: `slock` (M; ∈ suckless tools).

Utilities+:

- TLDR: `tealdeer` (`M`).

Utilities+:

- TLDR: [tealdeer](#) ([M](#)).
- offline Arch Wiki: [arch-wiki-docs](#) ([M](#)).

Utilities+:

- TLDR: [tealdeer](#) (M).
- offline Arch Wiki: [arch-wiki-docs](#) (M).
- fetching system information: [fastfetch](#) (M).

Applications+

Utilities+:

- TLDR: [tealdeer](#) (M).
- offline Arch Wiki: [arch-wiki-docs](#) (M).
- fetching system information: [fastfetch](#) (M).
- screenshot: [gnome-screenshot](#) (M).

Applications+

Utilities+:

- TLDR: [tealdeer](#) (M).
- offline Arch Wiki: [arch-wiki-docs](#) (M).
- fetching system information: [fastfetch](#) (M).
- screenshot: [gnome-screenshot](#) (M).
- fuzzy finder: [fzf](#) (M).

Applications+

Utilities+:

- TLDR: [tealdeer](#) (M).
- offline Arch Wiki: [arch-wiki-docs](#) (M).
- fetching system information: [fastfetch](#) (M).
- screenshot: [gnome-screenshot](#) (M).
- fuzzy finder: [fzf](#) (M).
- icon-like fonts: [nerd-fonts](#) (M; group).

Applications+

Utilities+:

- TLDR: [tealdeer](#) (M).
- offline Arch Wiki: [arch-wiki-docs](#) (M).
- fetching system information: [fastfetch](#) (M).
- screenshot: [gnome-screenshot](#) (M).
- fuzzy finder: [fzf](#) (M).
- icon-like fonts: [nerd-fonts](#) (M; group).
- sunrise, sunset, moonphase: [solunar2-git](#) (A).

Else: ?

Applications+

Utilities+:

- TLDR: [tealdeer](#) (M).
- offline Arch Wiki: [arch-wiki-docs](#) (M).
- fetching system information: [fastfetch](#) (M).
- screenshot: [gnome-screenshot](#) (M).
- fuzzy finder: [fzf](#) (M).
- icon-like fonts: [nerd-fonts](#) (M; group).
- sunrise, sunset, moonphase: [solunar2-git](#) (A).

Else: ?

My view

I do not miss anything, and have freedom.

A journey

Win
start



A journey

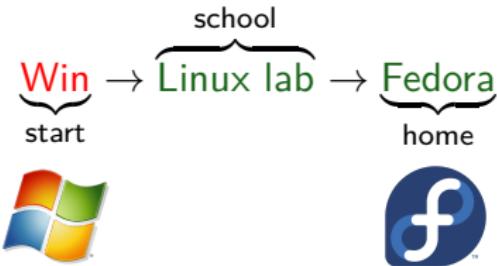
Win → Linux lab

start

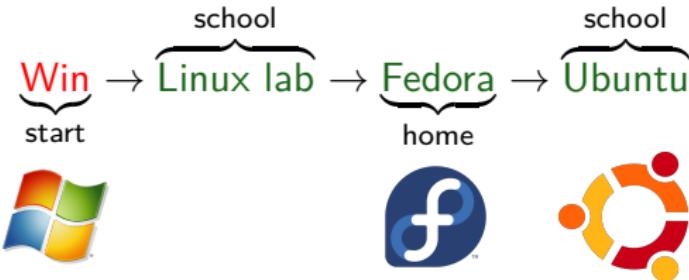
school



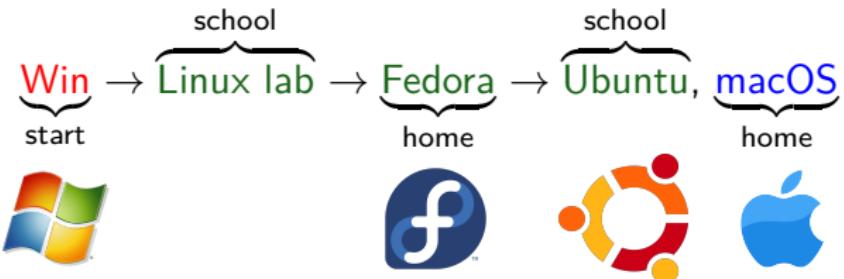
A journey



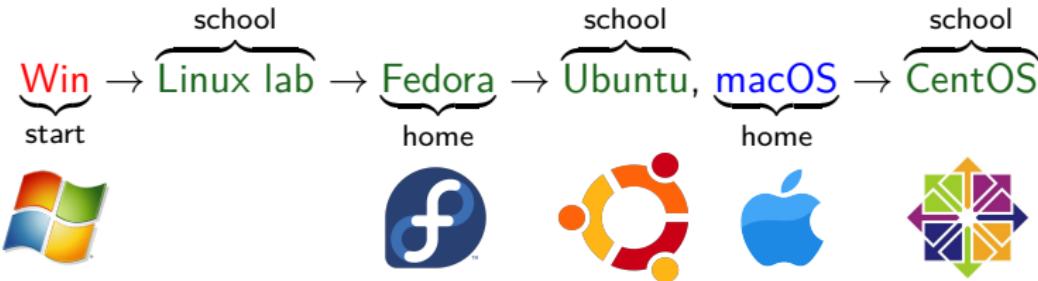
A journey



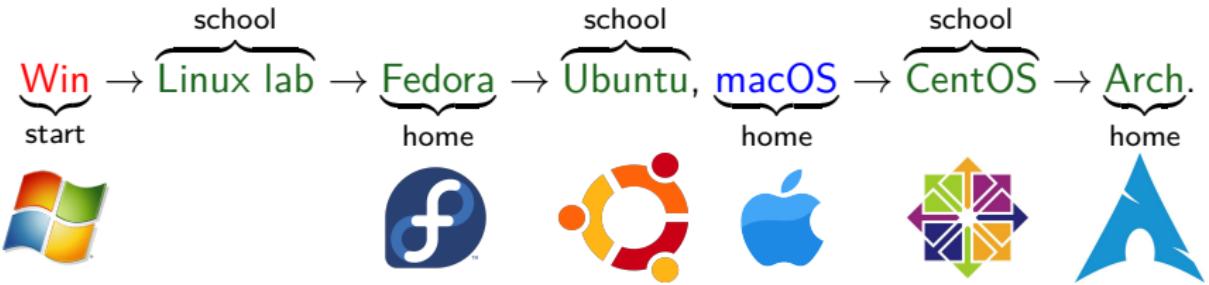
A journey



A journey



A journey



⇒ Currently

- laptop, dept-al server, research computing (cluster),
- mobile phone: PinePhone Pro,
- paper-like tablet ([Remarkable Paper Pro](#)),
- router (OpenWrt ∈ [Flint 2](#)),
- home automation ([Home Assistant](#)): reading about it.

Some fun (a 21Y Finish CS student)

From: Linus Benedict Torvalds

Date: Aug 25, 1991, 10:57:08 PM

Newsgroups: comp.os.mimix

Body:

Hello everybody ...

I'm doing a (free) operating system (just
a hobby, won't be big...)

Some fun: 30th birthday = 4 years ago.

From: Linus Benedict Torvalds
Date: Aug 25, 1991, 10:57:08 PM
Newsgroups: comp.os.mimix
Body:
Hello everybody ...

I'm doing a (free) operating system (just
a hobby, won't be big...)



#30YearsofLinux



Some fun: 30th birthday = 4 years ago.

From: Linus Benedict Torvalds
Date: Aug 25, 1991, 10:57:08 PM
Newsgroups: comp.os.mimix
Body:
Hello everybody ...

I'm doing a (free) operating system (just
a hobby, won't be big...)



#30YearsofLinux



Linus Torvalds (~now):



- Supercomputer world: 100% market share,
- Runs: from old laptops [OD_1 , YT_1 ; $1980=YT_2$] to top 500 supercomputers, even on a RISC-V laptop [YT]



- Supercomputer world: 100% market share,
- Runs: from old laptops [OD_1 , YT_1 ; $1980=YT_2$] to top 500 supercomputers, even on a RISC-V laptop [YT]



- At the heart of > 3 billion Android devices.

Linux today – continued

- Web-facing servers, Microsoft's own Azure cloud, game consoles (**SteamOS**), smart TVs, smart watches, Amazon Kindle, international space stations, ...

Linux today – continued

- Web-facing servers, Microsoft's own Azure cloud, game consoles (**SteamOS**), smart TVs, smart watches, Amazon Kindle, international space stations, ...
- Some slightly crazier examples:
 - Running Linux in a .pdf file [**OD**, **YT**; **article**],

- Web-facing servers, Microsoft's own Azure cloud, game consoles ([SteamOS](#)), smart TVs, smart watches, Amazon Kindle, international space stations, ...
- Some slightly crazier examples:
 - Running Linux in a .pdf file [[OD](#), [YT](#); [article](#)],
 - Booting Linux from Google Drive [[article](#)].

Linux today – continued

- Web-facing servers, Microsoft's own Azure cloud, game consoles (**SteamOS**), smart TVs, smart watches, Amazon Kindle, international space stations, ...
- Some slightly crazier examples:
 - Running Linux in a .pdf file [**OD**, **YT**; **article**],
 - Booting Linux from Google Drive [**article**].

One of the main secrets

free and open source ⇒ knowledge sharing ⇒ creativity can kick in ⇒ versatility!

In fact, Linux = **GNU/Linux**: Linus used the GNU development tools for his kernel, ...

- Late 1970s: companies started to spread proprietary software ⇒



- **GNU project** = **GNU is Not Unix**:

- Goal: write a UNIX-like operating system entirely of **free software**.
- Users are **legally free** (GPL)
 - ① to use,
 - ② to study,
 - ③ to modify, and
 - ④ to distribute the software.

In fact, Linux = **GNU/Linux**: Linus used the GNU development tools for his kernel, ...

- Late 1970s: companies started to spread proprietary software ⇒



- **GNU project** = **GNU is Not Unix**:

- Goal: write a UNIX-like operating system entirely of **free software**.
- Users are **legally free** (GPL)
 - ① to use,
 - ② to study,
 - ③ to modify, and
 - ④ to distribute the software.

user freedom (to make choices; free ≠ gratis)

In fact, Linux = **GNU/Linux**: Linus used the GNU development tools for his kernel, ...

- Late 1970s: companies started to spread proprietary software ⇒



- **GNU project** = **GNU is Not Unix**:

- Goal: write a UNIX-like operating system entirely of **free software**.
- Users are **legally free** (GPL)
 - ① to use,
 - ② to study,
 - ③ to modify, and
 - ④ to distribute the software.

user freedom (to make choices; free ≠ gratis)

- give computer **users freedom and control** in their use of their computers.

In fact, Linux = **GNU/Linux**: Linus used the GNU development tools for his kernel, ...

- Late 1970s: companies started to spread proprietary software ⇒



- **GNU project** = **GNU is Not Unix**:

- Goal: write a UNIX-like operating system entirely of **free software**.
- Users are **legally free** (GPL)
 - ① to use,
 - ② to study,
 - ③ to modify, and
 - ④ to distribute the software.

user freedom (to make choices; free ≠ gratis)

- give computer **users freedom and control** in their use of their computers.
- mass collaboration announced by **Richard Stallman** ('83; **his website**).

Richard (Matthew) Stallman: RMS – his hacker name;)



- Founder of the Free Software Foundation ('85) → resources,
 - non-profit organization to support the free software movement.

Richard (Matthew) Stallman: RMS – his hacker name;)



- Founder of the **Free Software Foundation** ('85) → **resources**,
 - non-profit organization to support the free software movement.
- Author of the **GNU GPL license** ('89): for the GNU project,



1st **copyleft** license

any derivative work must be distributed under the same terms.

Richard (Matthew) Stallman: RMS – his hacker name;)



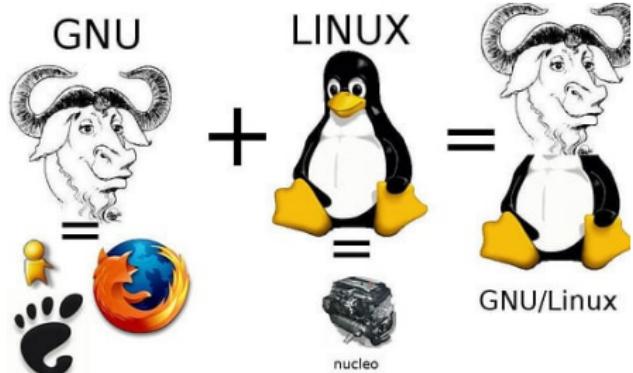
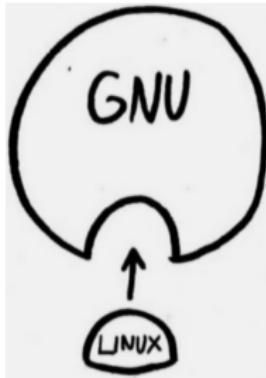
- Founder of the **Free Software Foundation** ('85) → **resources**,
 - non-profit organization to support the free software movement.
- Author of the **GNU GPL license** ('89): for the GNU project,



1st **copyleft** license

any derivative work must be distributed under the same terms.

- Creator of **GNU Emacs**: 'text editor' (**LISP** interpreter).



⇒

- ① **GNU utils:** high-level utilities.
- ② **Kernel:**
 - low-level 'stuff', written (mostly) in C, GPLv2,
 - manages the CPU, memory, device drivers, file system, ...

- Tux (the mascot of the Linux kernel) := Torvalds UniX.



- Tux (the mascot of the Linux kernel) := Torvalds UniX.



- Author of Git (GPLv2):
 - distributed version control system,
 - gold standard in collaborative coding efforts,
 - developed for the Linux kernel ('05),
 - # of lines in the code of Linux kernel: 40+ million (2025).



Free vs open

Free software (a.k.a. **FOSS**, libre software):

- goal: to respect user freedom and privacy.

to *not constrain* the user

Free vs open

Free software (a.k.a. **FOSS**, libre software):

- goal: to respect user freedom and privacy.

to *not constrain* the user

- free \ni open-source, but
 - free \neq open-source: text, video [OD, YT],
 - open-source code can 'spy' on you,
privacy matters [OD, YT].



Free vs right to repair

- my experience: battery replacement in Surface Pro = 600€,
- idea in 60s [OD,YT]: \$12.9 << \$1500 (repairing for 17Y@2025; ~iPhone)



Free vs right to repair

- my experience: battery replacement in Surface Pro = 600€,
- idea in 60s [OD,YT]: \$12.9 < \$1500 (repairing for 17Y@2025; ~iPhone),
- Free future hints:
 - System76: repairable laptops, Launch keyboard, Pop!_OS, Virgo (design),



Free vs right to repair

- my experience: battery replacement in Surface Pro = 600€,
- idea in 60s [OD,YT]: \$12.9 < \$1500 (repairing for 17Y@2025; ~iPhone),
- Free future hints:
 - System76: repairable laptops, Launch keyboard, Pop!_OS, Virgo (design),
 - Tuxedo: customizable, repairable PCs.



Free vs right to repair

- my experience: battery replacement in Surface Pro = 600€,
- idea in 60s [OD,YT]: \$12.9 < \$1500 (repairing for 17Y@2025; ~iPhone),
- Free future hints:
 - System76: repairable laptops, Launch keyboard, Pop!_OS, Virgo (design),
 - Tuxedo: customizable, repairable PCs.
 - a laptop initiative: frame.work ≈ anti-MacBook ⇐ designed to be easy to upgrade & repair.



- **community-driven**, **versatile**, **transparent**, **secure & private**, **modular**,
resource-efficient, **sustainable**.

- community-driven, versatile, transparent, secure & private, modular, resource-efficient, sustainable.
- specifically:
 - no force to
 - ① upgrade to the latest hardware

- community-driven, versatile, transparent, secure & private, modular, resource-efficient, sustainable.
- specifically:
 - no force to
 - ① upgrade to the latest hardware,
 - ② throw money out of the window (e.g., Win 11 Pro: £219.99)

- **community-driven, versatile, transparent, secure & private, modular, resource-efficient, sustainable.**
- specifically:
 - no force to
 - ① upgrade to the latest hardware,
 - ② throw money out of the window (e.g., Win 11 Pro: £219.99),
 - ③ have your entire life history saved (e.g., Win 11: Recall ∈ Copilot+ PCs [OD₁, YT₁; OD₂, YT₂; OD₃, YT₃; article]; mobile: Google & Apple push notifications → governments [OD, YT], collect data even in incognito mode or with apps disabled)

- **community-driven, versatile, transparent, secure & private, modular, resource-efficient, sustainable.**
- specifically:
 - no force to
 - ① upgrade to the latest hardware,
 - ② throw money out of the window (e.g., Win 11 Pro: £219.99),
 - ③ have your entire life history saved (e.g., Win 11: Recall ∈ Copilot+ PCs [OD₁, YT₁; OD₂, YT₂; OD₃, YT₃; article]; mobile: Google & Apple push notifications → governments [OD, YT], collect data even in incognito mode or with apps disabled),
 - ④ create accounts or watch dummy ads on the UI

- **community-driven, versatile, transparent, secure & private, modular, resource-efficient, sustainable.**
- specifically:
 - no force to
 - ① upgrade to the latest hardware,
 - ② throw money out of the window (e.g., Win 11 Pro: £219.99),
 - ③ have your entire life history saved (e.g., Win 11: Recall ∈ Copilot+ PCs [OD₁, YT₁; OD₂, YT₂; OD₃, YT₃; article]; mobile: Google & Apple push notifications → governments [OD, YT], collect data even **in incognito mode or with apps disabled**),
 - ④ create accounts or watch dummy ads on the UI,
 - ⑤ go with the **trust us bro** security/privacy guarantee.

- specifically – continued:
 - significant storage reduction ⇐ code sharing.

- specifically – continued:
 - significant storage reduction ⇐ code sharing.
 - no **intentional slow down of your system** (e.g., iPhone).

- specifically – continued:
 - significant storage reduction ⇐ code sharing.
 - no **intentional slow down of your system** (e.g., iPhone).
 - standard for computing clusters (example: **slurm**).



Versatility ⇒

Various distros (tree):

- there have been > 1000 distributions,
- currently (Sept. 26, 2025): 367 distributions



Versatility ⇒

Various distros (tree):

- there have been > 1000 distributions,
- currently (Sept. 26, 2025): 367 distributions,
- but minor differences.



Primary choice to make

point release vs rolling release.

Point release model: Windows

- Example:
 - **Windows:** 3.0, 3.1x, 95, 98, Me, NT, 2000, XP, Vista, 7, 8, 10, 11.

Point release model: Windows, macOS

- Example:
 - **Windows:** 3.0, 3.1x, 95, 98, Me, NT, 2000, XP, Vista, 7, 8, 10, 11.
 - **macOS:** Cheetah, Puma, Jaguar, Panther, Tiger, Leopard, Snow Leopard, Lion, Mountain Lion, Mavericks, Yosemite, El Capitan, Sierra, High Sierra, Mojave, Catalina, Big Sur, Monterey, Ventura, Sonoma, Sequoia, Tahoe.

Point release model: Windows, macOS

- Example:
 - Windows: 3.0, 3.1x, 95, 98, Me, NT, 2000, XP, Vista, 7, 8, 10, 11.
 - macOS: Cheetah, Puma, Jaguar, Panther, Tiger, Leopard, Snow Leopard, Lion, Mountain Lion, Mavericks, Yosemite, El Capitan, Sierra, High Sierra, Mojave, Catalina, Big Sur, Monterey, Ventura, Sonoma, Sequoia, Tahoe.
- Properties:
 - ① occasional **big** changes,
 - ② **end-of-life** date!

Point release: Linux distributions



(Debian →) Ubuntu → Pop!_OS; Fedora; openSUSE.

Point release: Linux distributions



(Debian →) Ubuntu → Pop!_OS; Fedora; openSUSE.

- **Ubuntu:**

- developer: Canonical Ltd.,
- released every six months, LTS every 2 years.

Point release: Linux distributions



(Debian →) Ubuntu → Pop!_OS; Fedora; openSUSE.

- **Ubuntu:**

- developer: Canonical Ltd.,
- released every six months, LTS every 2 years.

- **Pop!_OS:**

- ≈ Ubuntu with customized Gnome (**COSMIC**),
- maintained by **System76**.

Point release: Linux distributions



(Debian →) Ubuntu → Pop!_OS; Fedora; openSUSE.

- **Ubuntu:**
 - developer: Canonical Ltd.,
 - released every six months, LTS every 2 years.
- **Pop!_OS:**
 - ≈ Ubuntu with customized Gnome (**COSMIC**),
 - maintained by **System76**.
- **Fedora:**
 - upstream source for Red Hat Enterprise Linux (developed by **Red Hat**),
 - released every 6 months.

Point release: Linux distributions



(Debian →) Ubuntu → Pop!_OS; Fedora; openSUSE.

- **Ubuntu:**
 - developer: Canonical Ltd.,
 - released every six months, LTS every 2 years.
- **Pop!_OS:**
 - ≈ Ubuntu with customized Gnome (**COSMIC**),
 - maintained by **System76**.
- **Fedora:**
 - upstream source for Red Hat Enterprise Linux (developed by **Red Hat**),
 - released every 6 months.
- **openSUSE: Leap,**
 - released 1×/year.

Rolling release: Linux distributions

- Arch Linux:



- one-time installation with continuous upgrades,
- lightweight and flexible,
- follows the keep it simple (**KISS**) principle,
- designed to teach its user.

Rolling release: Linux distributions

- other examples: [openSUSE Tumbleweed](#), [Gentoo](#).



Point release vs rolling release

point	rolling
always up-to-date	+
(new software features, bug fixes, security patches)	
supports even very new hardware	+
more secure	+
no need to reinstall it	+
requires semi-decent internet bandwidth	-
less suited for servers (where stability is max-ed)	-

My choice: Arch (released in 2002)

- ① rolling release.
- ② great package manager (pacman),
 - fast,
 - allows parallel downloading.



[Package managers handle dependencies.]

My choice: Arch (released in 2002)

- ① rolling release.
- ② great package manager (pacman),
 - fast,
 - allows parallel downloading.



- [Package managers handle dependencies.]
- ③ superb documentation ([Arch Wiki](#)):
 - base reference even for other distributions;)
 - DRY (don't repeat yourself) principle [[YT](#), [OD](#)].
 - even offline readable [[OD](#), [YT](#)] and searchable [[OD](#), [YT](#)].

My choice: Arch (released in 2002)

- ① rolling release.
- ② great package manager (pacman),
 - fast,
 - allows parallel downloading.



[Package managers handle dependencies.]

- ③ superb documentation ([Arch Wiki](#)):
 - base reference even for other distributions;)
 - DRY (don't repeat yourself) principle [[YT](#), [OD](#)].
 - even offline readable [[OD](#), [YT](#)] and searchable [[OD](#), [YT](#)].
- ④ excellent software availability:
 - main: 15.5K+

My choice: Arch (released in 2002)

- ① rolling release.
- ② great package manager (pacman),
 - fast,
 - allows parallel downloading.



[Package managers handle dependencies.]

- ③ superb documentation ([Arch Wiki](#)):
 - base reference even for other distributions;)
 - DRY (don't repeat yourself) principle [[YT](#), [OD](#)].
 - even offline readable [[OD](#), [YT](#)] and searchable [[OD](#), [YT](#)].
- ④ excellent software availability:
 - main: 15.5K+, AUR: 100K+ packages.

Both are searchable.

Installation

- ➊ Download, check, **burn the installation .iso to a USB stick**, leave the stick in your machine, reboot.



Installation

- ➊ Download, check, **burn the installation .iso to a USB stick**, leave the stick in your machine, reboot.



- ➋ Enter into the 'BIOS' (by pressing Esc, F1/F2/...):
 - disable Secure boot,
 - choose the boot medium to be the USB stick.

Installation

- ➊ Download, check, **burn the installation .iso to a USB stick**, leave the stick in your machine, reboot.



- ➋ Enter into the 'BIOS' (by pressing Esc, F1/F2/...):
 - disable Secure boot,
 - choose the boot medium to be the USB stick.
- ➌ Boot from the USB stick.

Installation

- ➊ Download, check, **burn the installation .iso to a USB stick**, leave the stick in your machine, reboot.



- ➋ Enter into the 'BIOS' (by pressing Esc, F1/F2/...):
 - disable Secure boot,
 - choose the boot medium to be the USB stick.
- ➌ Boot from the USB stick.
- ➍ Follow the instructions.

Downloading note

- .iso size:
 - 15.8 GB (MacOS Tahoe) – for comparison.

Downloading note

- .iso size:
 - 5.4 GB (Windows 11) – for comparison.
 - 15.8 GB (MacOS Tahoe) – for comparison.

Downloading note

- .iso size:

- 5.9 GB (Ubuntu),
- 5.4 GB (Windows 11) – for comparison.
- 15.8 GB (MacOS Tahoe) – for comparison.

Downloading note

- .iso size:
 - 2.56 GB (Pop!_OS),
 - 5.9 GB (Ubuntu),
 - 5.4 GB (Windows 11) – for comparison.
 - 15.8 GB (MacOS Tahoe) – for comparison.

Downloading note

- .iso size:

- 2.2 GB (Fedora),
- 2.56 GB (Pop!_OS),
- 5.9 GB (Ubuntu),
- 5.4 GB (Windows 11) – for comparison.
- 15.8 GB (MacOS Tahoe) – for comparison.

Downloading note

- .iso size:

- 1.4 GB (Arch),
- 2.2 GB (Fedora),
- 2.56 GB (Pop!_OS),
- 5.9 GB (Ubuntu),
- 5.4 GB (Windows 11) – for comparison.
- 15.8 GB (MacOS Tahoe) – for comparison.

Downloading note

- .iso size:
 - 1.4 GB (Arch),
 - 2.2 GB (Fedora),
 - 2.56 GB (Pop!_OS),
 - 5.9 GB (Ubuntu),
 - 5.4 GB (Windows 11) – for comparison.
 - 15.8 GB (MacOS Tahoe) – for comparison.
- downloading:
 - **http; torrent**: this can be faster (\Leftarrow sharing).



Notes on the boot process — a bit technical

- ① system's **firmware** (such as **BIOS/UEFI/Coreboot/Libreboot**) $\xrightarrow{\text{starts}}$
- ② **bootloader** (such as GRUB \Leftarrow GNU; **features & others**) $\xrightarrow{\text{loads}}$
- ③ the **kernel** (your operating system).

In practice:

Notes on the boot process — a bit technical

- ① system's **firmware** (such as **BIOS/UEFI/Coreboot/Libreboot**) $\xrightarrow{\text{starts}}$
- ② **bootloader** (such as GRUB \Leftarrow GNU; **features & others**) $\xrightarrow{\text{loads}}$
- ③ the **kernel** (your operating system).

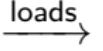
In practice:

- **firmware**:

- probes for hardware, simple health checks,
- it has a UI accessible with a magic key (Esc, F1/F2/...),
- allows you to designate a boot device (USB/hard/CD/DVD drive, ...),
- consults the GPT[†] partition table to identify the **ESP**[‡], and launches the target application (typically the bootloader).

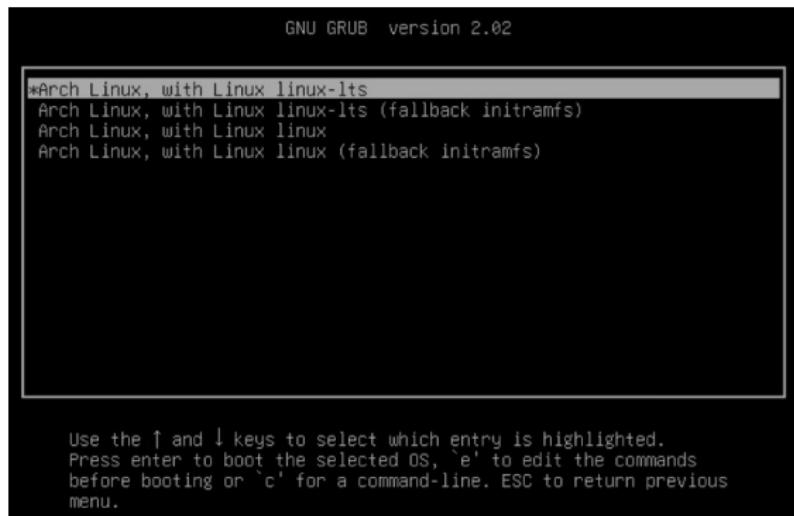
[†]no chat 😊, [‡]EFI System Partition.

Notes on the boot process — a bit technical

- ① system's **firmware** (such as **BIOS/UEFI/Coreboot/Libreboot**)  starts
- ② **bootloader** (such as GRUB ⇐ GNU; **features & others**)  loads
- ③ the **kernel** (your operating system).

In practice:

- **bootloader:**
 - gives a menu on which kernel / operating system to invoke.



Instructions: for Arch – scary;)

- ① Step-by-step text guide (official one).
- ② Video guide:
 - vid₁ [OD, YT]: UEFI; check the YouTube comments as well!
 - vid₂ [YT]: BIOS, UEFI, UEFI-LVM-LUKS.

Instructions: for Arch – scary;)

- ① Step-by-step text guide (official one).
- ② Video guide:
 - vid₁ [OD, YT]: UEFI; check the YouTube comments as well!
 - vid₂ [YT]: BIOS, UEFI, UEFI-LVM-LUKS.

Definitions

- firmware: BIOS (very old machine), UEFI (semi-new computer).
- partition table: BIOS ⇒ MBR (a.k.a. DOS, MS-DOS); UEFI ⇒ GPT.
- LVM: adjustable layout, LUKS: encryption.
- LUKS: your data can't be read even if your laptop is stolen.

Instructions: BIOS or UEFI

BIOS:



UEFI:



Installation hints

- ① use ethernet: faster.

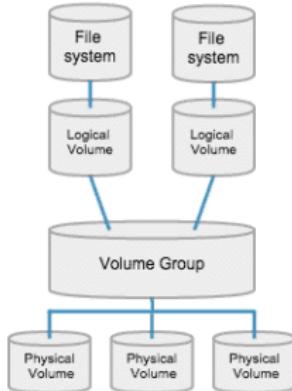


Installation hints

- ① use ethernet: faster.



- ② start simply: no encryption, no LVM.



Installation hints

- ② start simply – elaborated (nerdness-level dependent):
 - ① Live media/USB/image (Fedora, Ubuntu):
 - .iso writing, hardware support check, quick look at the system ✓,
 - slower than SSD.

Installation hints

- ② start simply – elaborated (nerdness-level dependent):
 - ① Live media/USB/image (Fedora, Ubuntu):
 - .iso writing, hardware support check, quick look at the system ✓,
 - slower than SSD.
 - ② graphical installer (Fedora, Ubuntu).

Installation hints

- ② start simply – elaborated (nerdness-level dependent):
 - ① Live media/USB/image (Fedora, Ubuntu):
 - .iso writing, hardware support check, quick look at the system ✓,
 - slower than SSD.
 - ② graphical installer (Fedora, Ubuntu).
 - ③ command line installer (Arch):
 - UEFI, no LVM, no LUKS: basic understanding,

Installation hints

② start simply – elaborated (nerdness-level dependent):

① Live media/USB/image (Fedora, Ubuntu):

- .iso writing, hardware support check, quick look at the system ✓,
- slower than SSD.

② graphical installer (Fedora, Ubuntu).

③ command line installer (Arch):

- UEFI, no LVM, no LUKS: basic understanding,
- UEFI, LVM, LUKS: slightly deeper understanding.

④ NixOS:

- declarative approach based on Nix [YT] ⇒ reproducibility,
- can be pretty neat (in the future).

Installation hints

② start simply – elaborated[†] ([nerdness-level dependent](#)):

① [Live media/USB/image](#) (Fedora, Ubuntu):

- .iso writing, hardware support check, quick look at the system ✓,
- slower than SSD.

② [graphical installer](#) (Fedora, Ubuntu).

③ [command line installer](#) (Arch):

- UEFI, no LVM, no LUKS: basic understanding,
- UEFI, LVM, LUKS: slightly deeper understanding.

④ [NixOS](#):

- declarative approach based on Nix [[YT](#)] ⇒ reproducibility,
- can be pretty neat (in the future).

[†]Start with a DE before a WM.

Installation hints – continued

③ partition formatting:

- **ext4**: more settled – my choice,



- **btrfs**:
 - modern alternative,
 - supports compression ⇒ less space, increased storage lifespan,
 - copy-on-write ⇒ consistency even in case of power loss,
 - snapshot feature,
 - limited LUKS support.
- ⇒ It is worth keeping an eye on it!

Installation hints – continued

- ④ kernel (stable), LTS kernel (longterm) [others]:
 - stable: maintained until the next stable release,
 - LTS: maintained for a few extra years,
 - good to have both: flexibility.

Installation hints – continued

④ kernel (stable), LTS kernel (longterm) [others]:

- stable: maintained until the next stable release,
- LTS: maintained for a few extra years,
- good to have both: flexibility.

⑤ swap:

- helps if RAM is exhausted (but slower, $\times 1000!$); size recommendations.
- 2 types:
 - ① swap partition: often preferred,
 - ② swap file: easier to resize, but less tested.

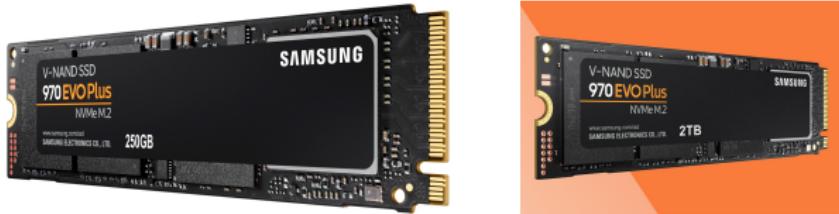
Installation hints – continued

- ⑥ good boot time (~ 11s):
 - SSD matters: Samsung 970 EVO Plus ← my choice (for laptop).



Installation hints – continued

- ⑥ good boot time (~ 11s):
 - SSD matters: Samsung 970 EVO Plus ← my choice (for laptop).



- ⑦ Use a spare drive (to avoid the wrestling of the op. systems),
 - example: (used) Thinkpad → T480 [YT]: low-budget, flexible.

Installation hints – continued

- ⑧ Create a normal user (beyond the root; \in wheel; sudo).

Installation hints – continued

- ⑧ Create a normal user (beyond the root; \in wheel; sudo).
- ⑨ Log your installation, usage, information sources (e.g. by Vimwiki)!

Installation hints – continued

- ⑧ Create a normal user (beyond the root; ∈ wheel; sudo).
- ⑨ Log your installation, usage, information sources (e.g. by Vimwiki)!
- ⑩ Post-installation:
 - think in terms of tasks not software, and use the native applications.
 - a weekly system update can be healthy.

Desktop environments (DE)

- Desktop environments:
 - [windows manager](#), and
 - a bundle of applications (calendar, image viewer, file manager, . . .).

Desktop environments (DE)

- Desktop environments:
 - **windows manager**, and
 - a bundle of applications (calendar, image viewer, file manager, . . .).
- You
 - are free to choose it, and can have multiple ones,
 - get workspaces.

Desktop environments (DE)

- Desktop environments:
 - [windows manager](#), and
 - a bundle of applications (calendar, image viewer, file manager, . . .).
- You
 - are free to choose it, and can have multiple ones,
 - get workspaces.
- Some popular choices: [KDE Plasma](#), [GNOME](#), [Xfce](#).

Desktop environments (DE)

- Desktop environments:
 - [windows manager](#), and
 - a bundle of applications (calendar, image viewer, file manager, . . .).
- You
 - are free to choose it, and can have multiple ones,
 - get workspaces.
- Some popular choices: [KDE Plasma](#), [GNOME](#), [Xfce](#).

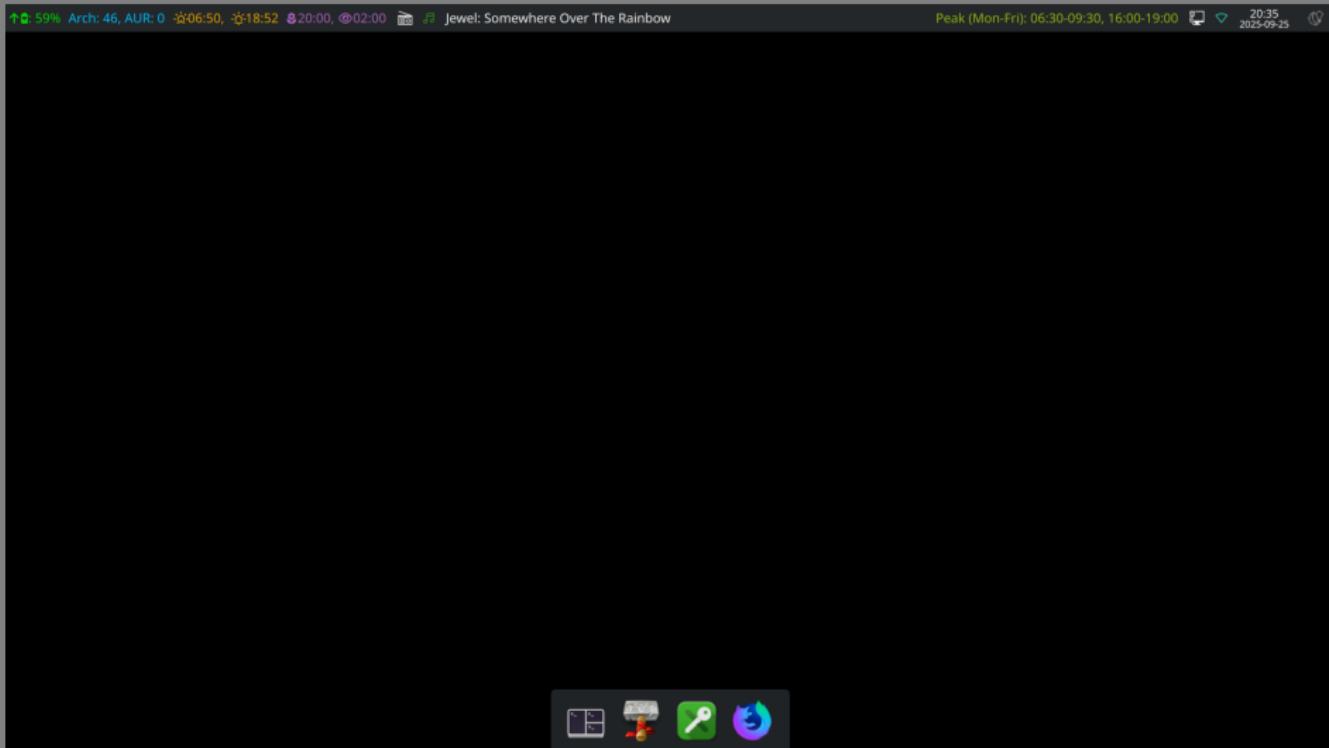
Examples follow

DE: KDE Plasma

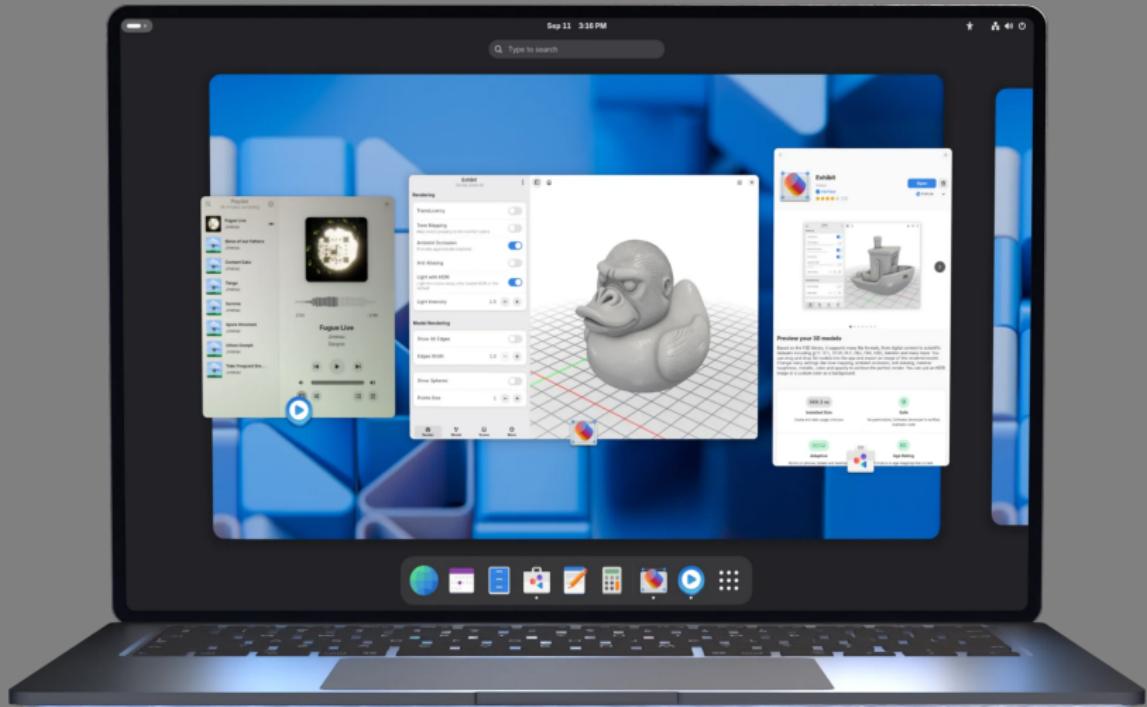


spec.
→ Win XP [YT] & Win 11 mimicing [OD, YT].

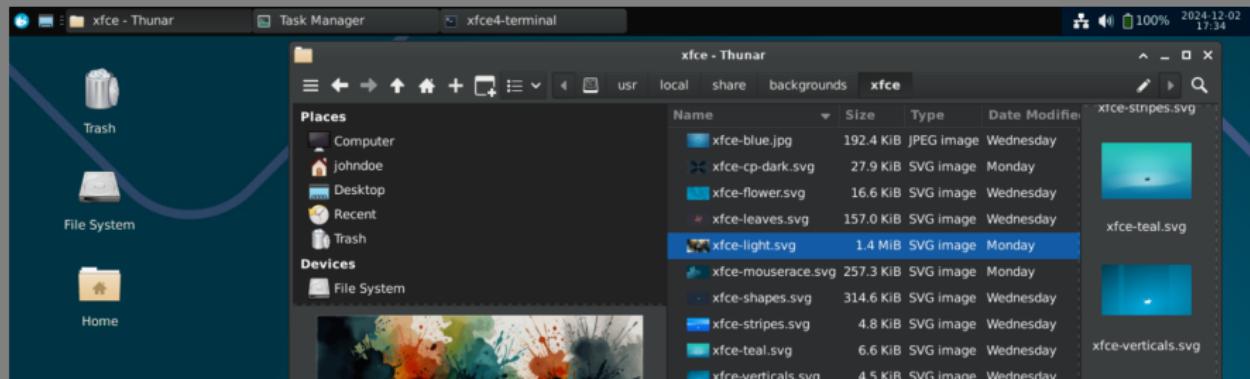
DE: KDE Plasma – my desktop



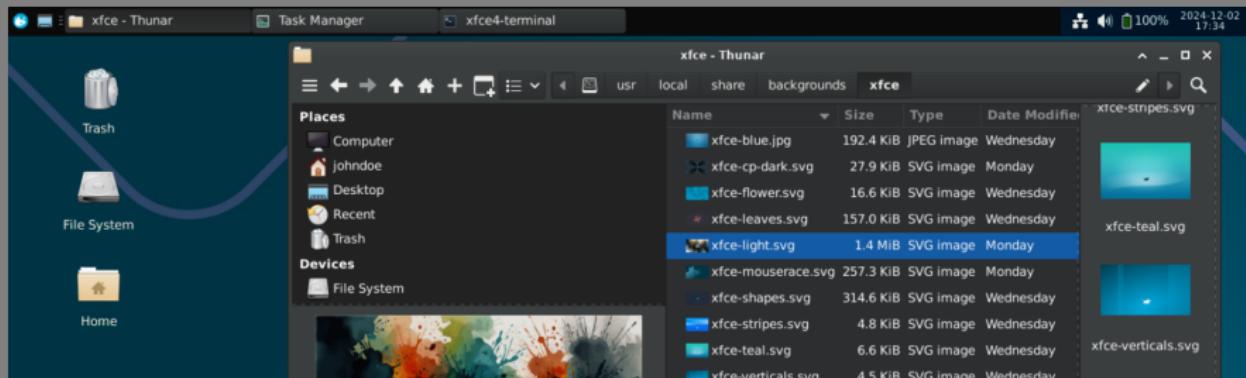
DE: GNOME



DE: Xfce



DE: Xfce



Extra DE inspiration

- [unixporn](#) (screenshots),
- [COSMIC DE](#): worth keeping an eye on it! (Wayland-based)



Window managers (WM)

- It allows handling windows (open, close, min/max-ze, move, resize, . . .).
- It can be part of a DE or standalone.
- Idea: WMs can be even snappier than DEs.

Window managers (WM)

- It allows handling windows (open, close, min/max-ze, move, resize, . . .).
- It can be part of a DE or standalone.
- Idea: WMs can be even snappier than DEs.
- 3 types:
 - ① **stacking** (a.k.a. floating),
 - ② **tiling**: non-overlapping windows,
 - ③ dynamic: allows switching between **tiling** and **floating** layout.

- Stacking:
 - KWin → KDE,
 - Mutter → GNOME,
 - Xfwm → Xfce.

- Stacking:
 - KWin → KDE,
 - Mutter → GNOME,
 - Xfwm → Xfce.
- Tiling:
 - i3.

- Stacking:
 - KWin → KDE,
 - Mutter → GNOME,
 - Xfwm → Xfce.
- Tiling:
 - i3.
- Dynamic:
 - Qtile:
 - it uses Python;)
 - various (77) **widgets**.

- Stacking:
 - KWin → KDE,
 - Mutter → GNOME,
 - Xfwm → Xfce.
- Tiling:
 - i3.
- Dynamic:
 - Qtile:
 - it uses Python;)
 - various (77) widgets.

Examples follow

WM: i3

```
main.c ("/i3/src") - VIM
}
/* Set up i3 specific atoms like I3_SOCKET_PATH and I3_CONFIG_PATH */
x_set_i3_atoms();

struct ev_io *xcb_watcher = malloc(sizeof(struct ev_io));
struct ev_io *xkb = malloc(sizeof(struct ev_io));
struct ev_check *xcb_check = malloc(sizeof(struct ev_check));
struct ev_prepare *xcb_prepare = malloc(sizeof(struct ev_prepare));

ev_io_init(xcb_watcher, xcb_get_event, xcb_get_file_descriptor(conn), EV_READ);
ev_io_start(main_loop, xcb_watcher);

if (xkb_supported) {
    ev_io_init(xkb, xkb_get_event, ConnectionNumber(xkbfd), EV_READ);
    ev_io_start(main_loop, xkb);
}

/* Flush the buffer so that libev can properly get new events */
MFflush(xkbfd);
}

ev_check_init(xcb_check, xcb_check_db);
ev_check_start(main_loop, xcb_check);

ev_prepare_init(xcb_prepare, xcb_prepare_db);
ev_prepare_start(main_loop, xcb_prepare);

xcb_flush(conn);

manage_existing_windows(root);

if ((disable_signal_handler)
    setuse_signal_handler();

/* Ignore SIGPIPE to survive errors when an IPC client disconnects
 * while we are sending him a message */
signal(SIGPIPE, SIG_IGN);

/* Autostarting exec-lines */
if (autostart) {
    struct Autostart *exec;
    TAILQ_FOREACH(exec, autostarts, autostarts) {
        LOG("auto-starting %s\n", exec->command);
        start_application(exec->command);
    }
}

/* Autostarting exec_always_lines */
struct Autostart *exec_always;
TAILQ_FOREACH(exec_always, autostarts_always, autostarts_always) {
    LOG("auto-starting (always!) %s\n", exec_always->command);
    start_application(exec_always->command);
}

ev_loops(main_loop, 0);
src/main.c
464,9      992
```

x200: xplayer 902/E05.avi

MP4Player



x200: git log

```
commit b8774212b3802badb615418f45d85566fcfc149b
Author: Michael Steapelberg <michael@stapelberg.de>
Date: Sun Jul 17 22:08:00 2011 +0200
```

 Add missing function prototype for strndup on Darwin (Thanks Marcus)

```
commit 3da3e691063f7c47f09461bb948f48abeb6fcf149b
Author: Michael Steapelberg <michael@stapelberg.de>
Date: Sun Jul 17 15:21:57 2011 +0200
```

 i3-config-wizard: use fgetln on Darwin, use strndup from FreeBSD on Darwin (Thanks Marcus)

```
commit 02dfb3e091a759d2152919779f6d00929a71cc
Author: Michael Steapelberg <michael@stapelberg.de>
Date: Sun Jul 17 15:18:45 2011 +0200
```

 use memmem and strndup from FreeBSD on Darwin (Thanks Marcus)

```
commit fe563ad6995eff95e2019bf8bba0a72e21d133b
Author: Michael Steapelberg <michael@stapelberg.de>
Date: Sun Jul 17 15:17:24 2011 +0200
```

 makefile: link -liconv on Darwin (Thanks Marcus)

```
commit 7512f633a79c290f4e60287fb2bcd689f025b42
Author: Michael Steapelberg <michael@stapelberg.de>
Date: Fri Jul 15 15:21:39 2011 +0200
[]
```

1 2 3 4

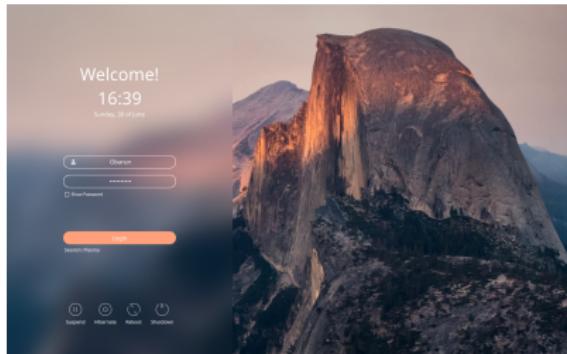
2001:4d80:100e:23:2f:16ff:fe1a:f9b0 | 2.3 GB | DHCP no | VPNs no | Us down | Et: 192.168.1.42 (1000 Mbit/s) | MTU 74, ISS: 1#] | 01s | 0,03 | 2011-07-22 15:02:32

WM: Qtile



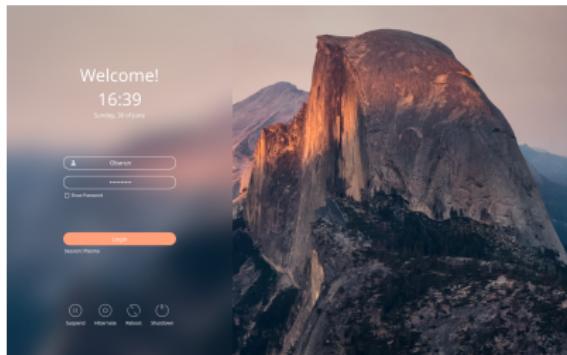
Login/display manager

- It gives graphical login – if you prefer not using/starting from tty;)
- Popular choices: (i) SDDM: tutorial [[OD](#),[YT](#)] ([inspiration+](#))



Login/display manager

- It gives graphical login – if you prefer not using/starting from tty;)
- Popular choices: (i) SDDM: tutorial [OD, YT] ([inspiration+](#)), (ii) Ly: tutorial [OD, YT]

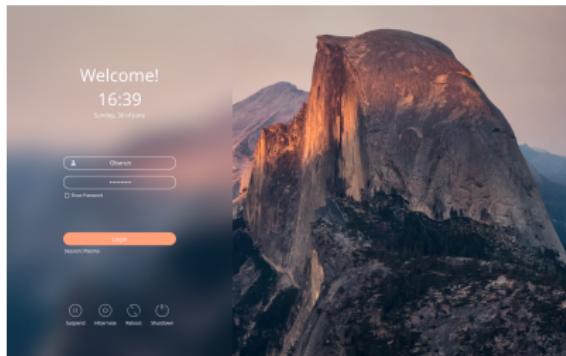


F1 shutdown F2 reboot

```
debian11
< GNOME on Xorg >
login:
password:
```

Login/display manager

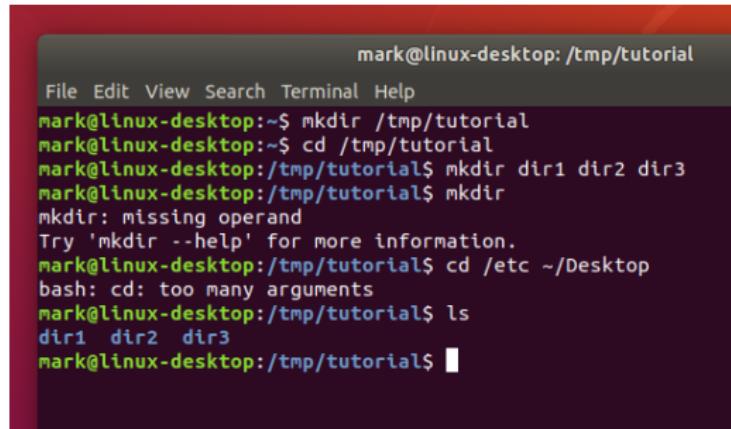
- It gives graphical login – if you prefer not using/starting from tty;)
- Popular choices: (i) SDDM: tutorial [OD, YT] ([inspiration+](#)), (ii) Ly: tutorial [OD, YT], (iii) GDM: tutorial [OD, YT] ([inspiration+](#)).



Terminal: used for instance @ Arch install

Command shell:

- like Jupyter notebook,
- interaction with the operating system,



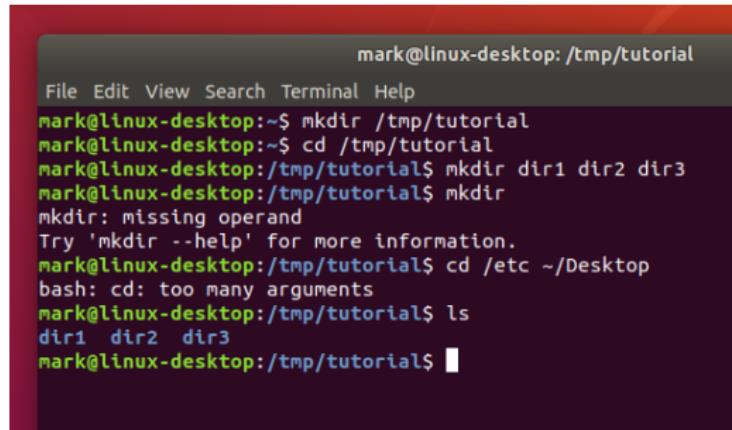
A screenshot of a terminal window titled "mark@linux-desktop: /tmp/tutorial". The window has a dark background and a light gray title bar. The terminal content shows a user interacting with the command line:

```
mark@linux-desktop:~$ mkdir /tmp/tutorial
mark@linux-desktop:~$ cd /tmp/tutorial
mark@linux-desktop:/tmp/tutorial$ mkdir dir1 dir2 dir3
mark@linux-desktop:/tmp/tutorial$ mkdir
mkdir: missing operand
Try 'mkdir --help' for more information.
mark@linux-desktop:/tmp/tutorial$ cd ~/Desktop
bash: cd: too many arguments
mark@linux-desktop:/tmp/tutorial$ ls
dir1 dir2 dir3
mark@linux-desktop:/tmp/tutorial$ █
```

Terminal: used for instance @ Arch install

Command shell:

- like Jupyter notebook,
- interaction with the operating system,



A screenshot of a terminal window titled "mark@linux-desktop: /tmp/tutorial". The window has a dark background and a light-colored text area. At the top, there's a menu bar with "File", "Edit", "View", "Search", "Terminal", and "Help". Below the menu, the terminal prompt is "mark@linux-desktop:~\$". The user runs several commands: "mkdir /tmp/tutorial", "cd /tmp/tutorial", "mkdir dir1 dir2 dir3", and "mkdir". The last "mkdir" command fails with the message "mkdir: missing operand" and "Try 'mkdir --help' for more information.". Then, the user tries to change directory to "/etc ~/Desktop" using "cd /etc ~/Desktop", which fails with "bash: cd: too many arguments". Finally, the user lists the contents of the directory with "ls", showing "dir1 dir2 dir3". The terminal ends with the prompt "mark@linux-desktop:/tmp/tutorial\$".

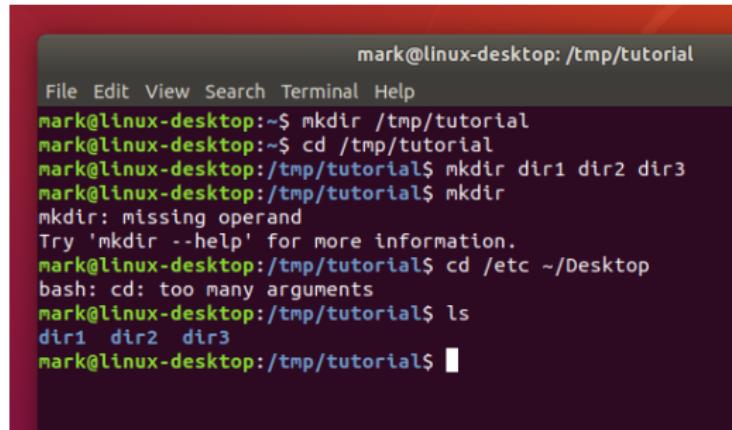
```
mark@linux-desktop: /tmp/tutorial
File Edit View Search Terminal Help
mark@linux-desktop:~$ mkdir /tmp/tutorial
mark@linux-desktop:~$ cd /tmp/tutorial
mark@linux-desktop:/tmp/tutorial$ mkdir dir1 dir2 dir3
mark@linux-desktop:/tmp/tutorial$ mkdir
mkdir: missing operand
Try 'mkdir --help' for more information.
mark@linux-desktop:/tmp/tutorial$ cd /etc ~/Desktop
bash: cd: too many arguments
mark@linux-desktop:/tmp/tutorial$ ls
dir1 dir2 dir3
mark@linux-desktop:/tmp/tutorial$
```

- Google Colab: !shellcommand

Terminal: used for instance @ Arch install

Command shell:

- like Jupyter notebook,
- interaction with the operating system,



A screenshot of a terminal window titled "mark@linux-desktop: /tmp/tutorial". The window has a dark background and a light-colored title bar. The terminal shows the following command-line session:

```
mark@linux-desktop:~$ mkdir /tmp/tutorial
mark@linux-desktop:~$ cd /tmp/tutorial
mark@linux-desktop:/tmp/tutorial$ mkdir dir1 dir2 dir3
mark@linux-desktop:/tmp/tutorial$ mkdir
mkdir: missing operand
Try 'mkdir --help' for more information.
mark@linux-desktop:/tmp/tutorial$ cd ~/Desktop
bash: cd: too many arguments
mark@linux-desktop:/tmp/tutorial$ ls
dir1 dir2 dir3
mark@linux-desktop:/tmp/tutorial$ █
```

- Google Colab: !shellcommand,
- **virtual console** = text terminal + login prompt (`ttyX` = `Ctrl+Alt+FX`, $X \in [7]$).

Command line (CLI) – continued

- Examples:

- \$ cd : change the current working directory,
- \$ ls : list directory content,
- \$ pwd : print the name of the current directory,
- \$ cp : copy files & directories,
- \$ mv : move or rename files and directories,
- \$ touch : create file,
- \$ mkdir : create directory,
- \$ man : manual page of a command.

Command line (CLI) – continued

- Examples:

- \$ cd : change the current working directory,
- \$ ls : list directory content,
- \$ pwd : print the name of the current directory,
- \$ cp : copy files & directories,
- \$ mv : move or rename files and directories,
- \$ touch : create file,
- \$ mkdir : create directory,
- \$ man : manual page of a command.

- shell := command line interpreter $\xrightarrow{\text{example}}$ Bash

Command line (CLI) – continued

- Examples:

- \$ cd : change the current working directory,
- \$ ls : list directory content,
- \$ pwd : print the name of the current directory,
- \$ cp : copy files & directories,
- \$ mv : move or rename files and directories,
- \$ touch : create file,
- \$ mkdir : create directory,
- \$ man : manual page of a command.

- shell := command line interpreter $\xrightarrow{\text{example}}$ Bash

⇒

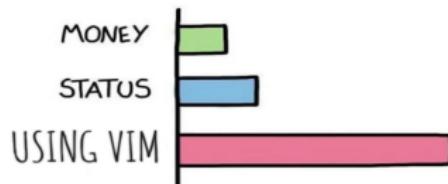
- shell/bash **scripting** [[tutorial](#)]
- lot of **automation** possibilities.

Superb text editor: Vim

- modal editor $\xrightarrow{\text{change}}$ `i` = input mode, `Esc` = command mode,
- highly customizable & efficient,
- keyboard-driven, language-like.
- cross-platform.



WHAT GIVES PEOPLE FEELINGS OF POWER



Vim: usage example

:q : quit ← most difficult;),
:w : write (save),
h,j,k,l : arrows (left, down, up, right)

Vim: usage example

:q : quit ← most difficult;),
:w : write (save),
h,j,k,l : arrows (left, down, up, right),
dw : delete word

Vim: usage example

:q : quit ← most difficult;),
:w : write (save),
h,j,k,l : arrows (left, down, up, right),
dw : delete word,
cw : change word

Vim: usage example

:q : quit ← most difficult;),
:w : write (save),
h,j,k,l : arrows (left, down, up, right),
dw : delete word,
cw : change word,
d3w : delete 3 words

Vim: usage example

:q : quit ← most difficult;),
:w : write (save),
h,j,k,l : arrows (left, down, up, right),
dw : delete word,
cw : change word,
d3w : delete 3 words,
di(: delete inside parentheses

Vim: usage example

:q : quit ← most difficult;),
:w : write (save),
h,j,k,l : arrows (left, down, up, right),
dw : delete word,
cw : change word,
d3w : delete 3 words,
di(: delete inside parentheses,
dd : delete line

Vim: usage example

:q : quit ← most difficult;),
:w : write (save),
h,j,k,l : arrows (left, down, up, right),
dw : delete word,
cw : change word,
d3w : delete 3 words,
di(: delete inside parentheses,
dd : delete line,
p : paste

Vim: usage example

:q : quit ← most difficult;),
:w : write (save),
h,j,k,l : arrows (left, down, up, right),
dw : delete word,
cw : change word,
d3w : delete 3 words,
di(: delete inside parentheses,
dd : delete line,
p : paste,
y : yank (copy)

Vim: usage example

:q : quit ← most difficult;),
:w : write (save),
h,j,k,l : arrows (left, down, up, right),
dw : delete word,
cw : change word,
d3w : delete 3 words,
di(: delete inside parentheses,
dd : delete line,
p : paste,
y : yank (copy),
yy : yank line, ...

Vim – continued (free ⇒)

- integration to browser, Jupyter notebook, ...
- evolution: vi → Vim → Neovim (community-developed),



- personal Wiki: vimwiki,
- tutorials: \$ vimtutor and

Name	Vids
ThePrimeagen	YT ₁ , YT ₂ , YT ₃ , YT ₄ , YT ₅ , YT ₆
Missing Semester	OD, YT
DistroTube	OD ₁ , YT ₁ ; OD ₂ , YT ₂
Ben Awad	YT

Hint: How to ask?

- Netiquette [OD,YT]; discussion → guide.
- DIY mentality:
 - the community is friendly and helps *if you put in effort,*
 - ⇒ **read & do your research first!**



Odysee (with LBRY) / YouTube channels: good educators

Name	Odysee	YouTube	
DistroTube	OD	YT	Linux
Learn Linux TV	—	YT	Linux
Brodie Robertson	OD	YT	Linux
Bread on Penguins	—	YT	Linux
ExplainingComputers	—	YT	IT
Eric Murphy	OD	YT	(less active nowadays [†])
EF - Linux Made Simple	OD	YT	(less active nowadays [†])
OldTechBloke	OD	YT	(less active nowadays [†])
Mental Outlaw	OD	YT	Linux, privacy
Luke Smith	OD	YT	(less active nowadays [†])
VeronicaExplains	—	YT	(less active [†])
MobileTechReview	—	YT	laptop & mobile reviews
Naomi Brockwell: NBTV	OD	YT	privacy
Louis Rossmann [‡]	OD	YT	right to repair

[†] but her/his videos are nice. [‡] FUTO's Guide to a Self Managed Life: part-1 [YT₁], part-2 [YT₂], wiki.

Ricing: Conky

- system monitor: CPU, memory, swap space, disk storage, temperature, processes, network interfaces, battery power, system messages, e-mail, ...

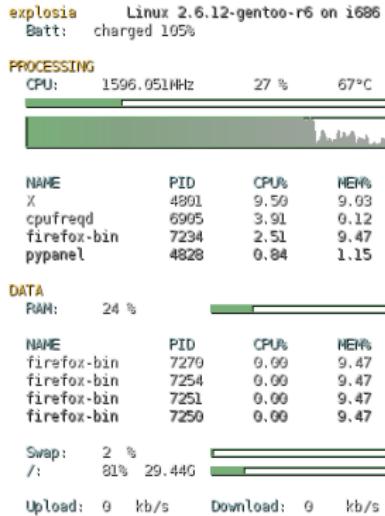
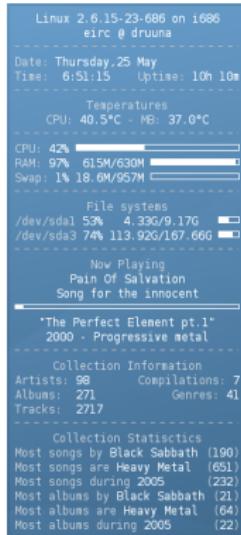
Ricing: Conky

- system monitor: CPU, memory, swap space, disk storage, temperature, processes, network interfaces, battery power, system messages, e-mail, ...
- Example ([inspiration+](#)):



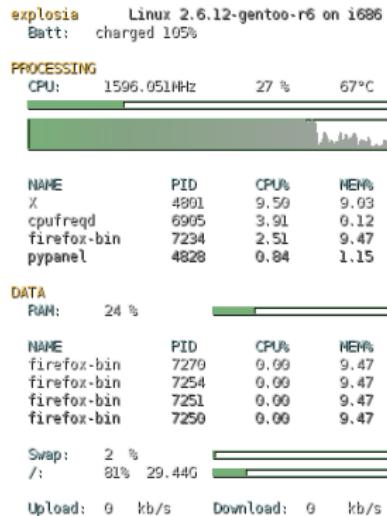
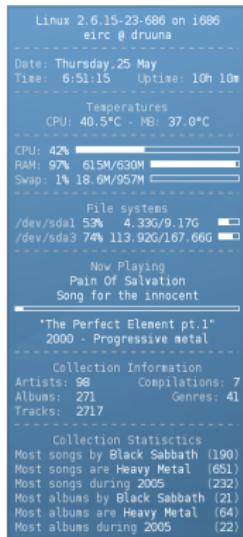
Ricing: Conky

- system monitor: CPU, memory, swap space, disk storage, temperature, processes, network interfaces, battery power, system messages, e-mail, ...
- Example ([inspiration+](#)):



Ricing: Conky

- system monitor: CPU, memory, swap space, disk storage, temperature, processes, network interfaces, battery power, system messages, e-mail, ...
- Example (inspiration+):



Ricing: Conky on desktop



Ricing: Conky on desktop

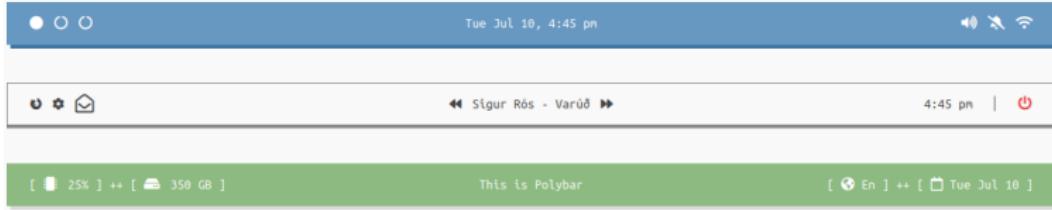


Ricing: Polybar ([link₁](#), [link₂](#))



- fast replacement of the status bar,
- date, time, keyboard layout, backlight, volume, MPD, network, CPU, ...

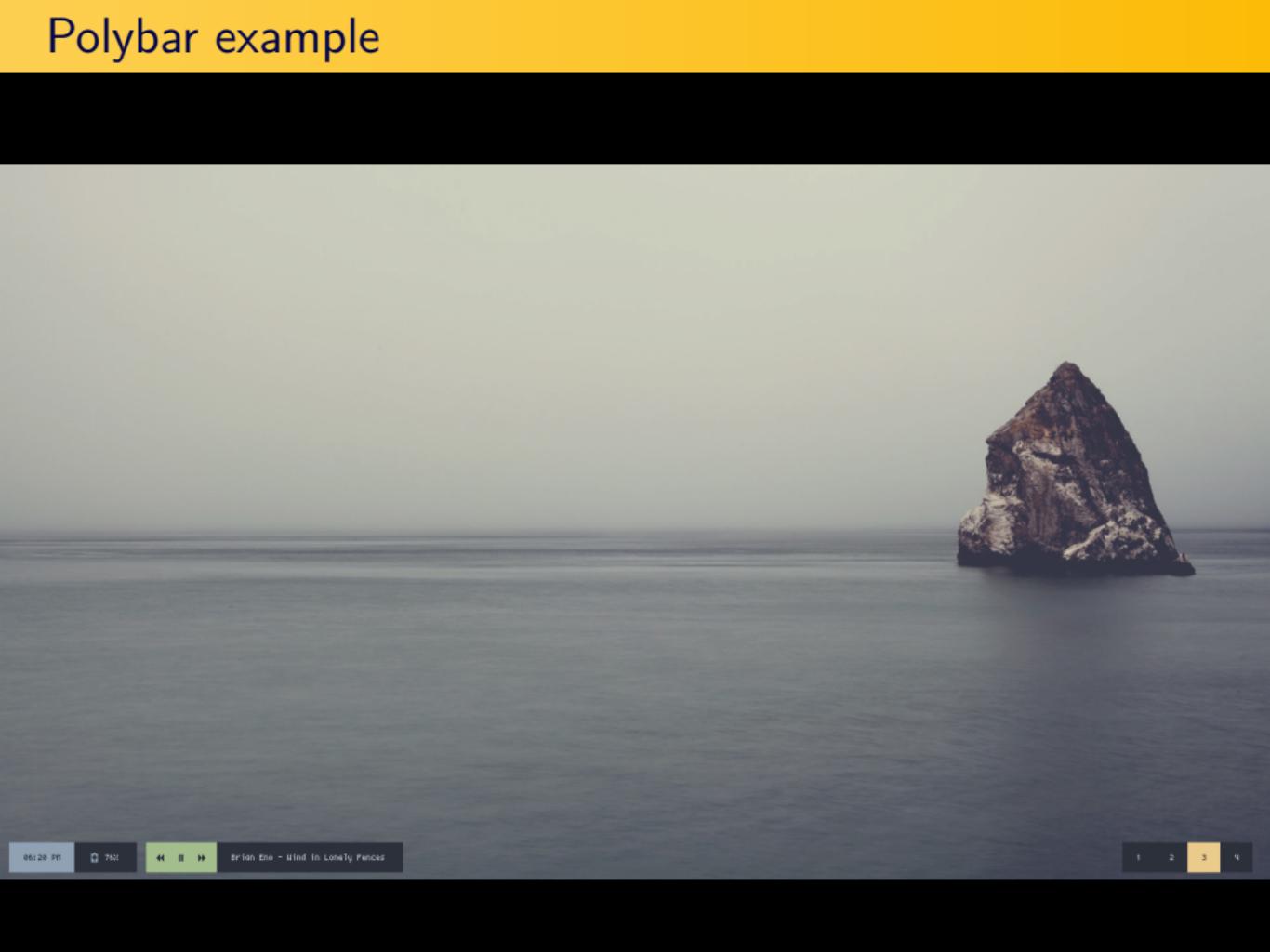
Ricing: Polybar ([link₁](#), [link₂](#))



- fast replacement of the status bar,
- date, time, keyboard layout, backlight, volume, MPD, network, CPU, ...

Example follows.

Polybar example



Compositors: for Xorg (a.k.a. X), for Wayland

- They can
 - add effects like transparency, animations or blur,
 - be standalone or built into the DE / WM.

Compositors: for Xorg (a.k.a. X), for Wayland

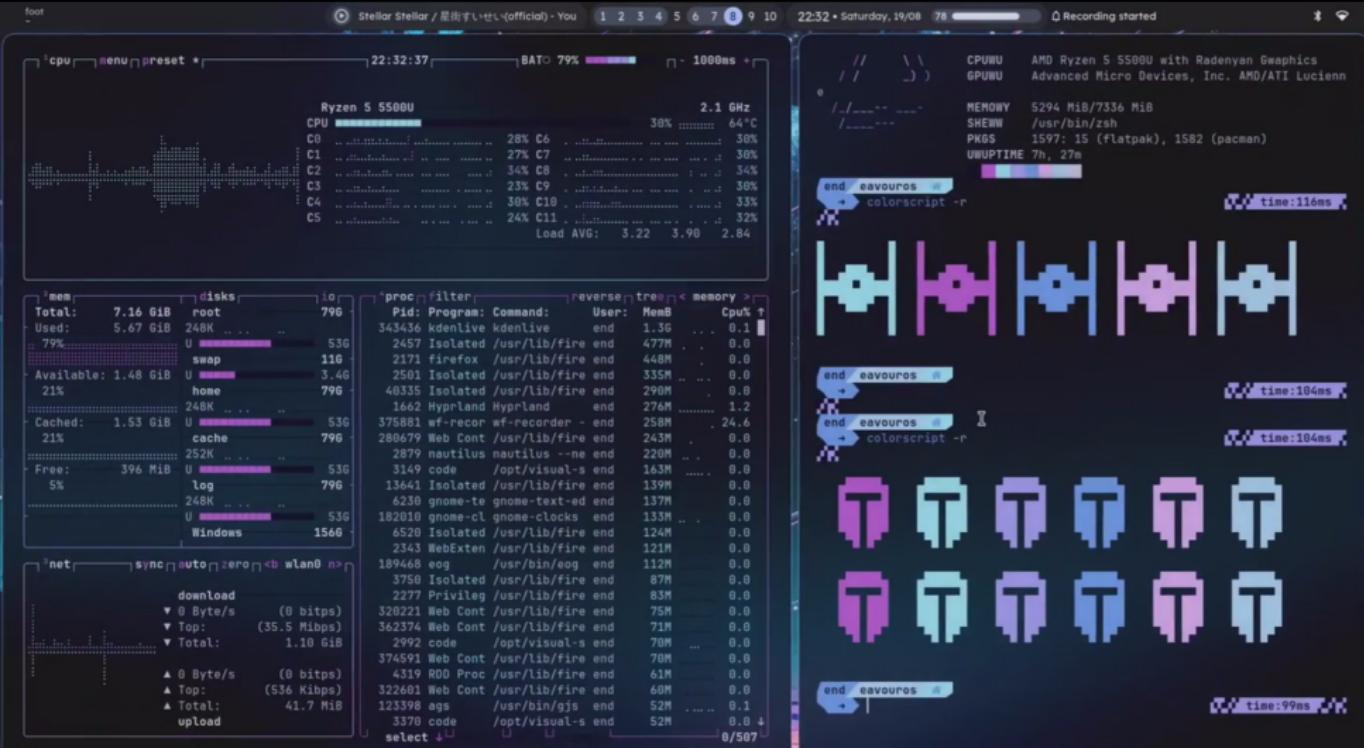
- They can
 - add effects like transparency, animations or blur,
 - be standalone or built into the DE / WM.
- Example: Picom, Hyprland, Wayfire. Related fun [OD,YT] with Gnome extensions.

Compositors: for Xorg (a.k.a. X), for Wayland

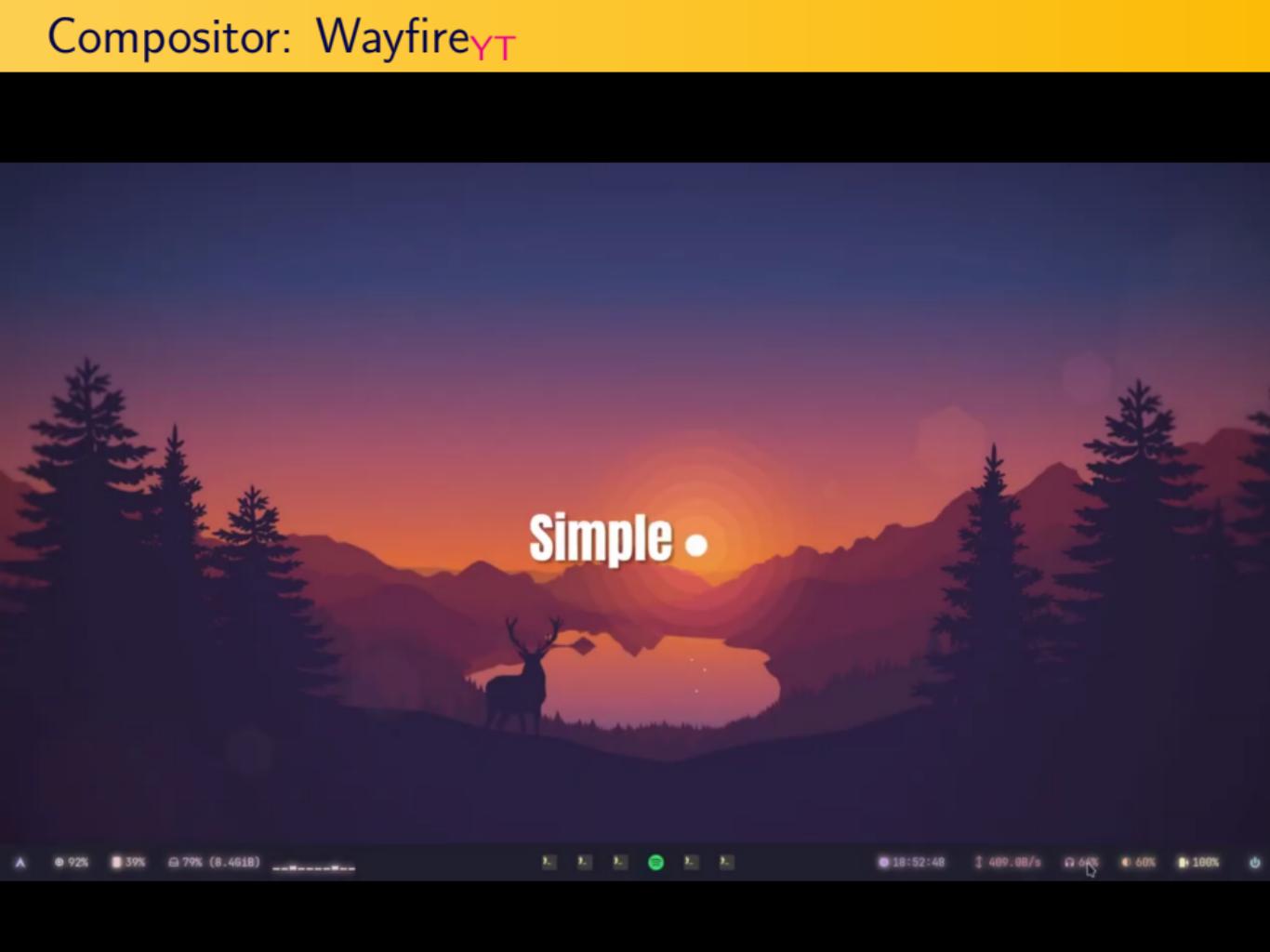
- They can
 - add effects like transparency, animations or blur,
 - be standalone or built into the DE / WM.
- Example: [Picom](#), [Hyprrland](#), [Wayfire](#). Related fun [OD,YT] with Gnome extensions.
- For Hyprrland: [website](#), [wiki](#); [demo](#) (screenshot):



Composer: Hyperlnd_{vid}



Composer: WayfireYT

A landscape illustration featuring a sunset with orange and yellow hues over a range of mountains. In the foreground, a deer stands on a grassy hill. The word "Simple" is overlaid in white, with a small black dot to its right.

Simple •

Linux phones (security & privacy; beta!)

① Librem 5:

- by Purism, running PureOS.



Linux phones

② PinePhone, PinePhone Pro:

- by Pine64,
- PinePhone Pro: [Wiki](#); various op. systems ⊇ Arch;)
- recipe: [YT₁](#), [YT₂](#), [YT₃](#), [YT₄](#), [YT₅](#).
- exploring: Arch with [Phosh](#) (SSH, VNC, ... ✓); [sxmo](#): looks exciting.



My choice (more stable and transparent communication)

③ To keep an eye on: [Liberux Nexx](#)!

Summary



- desktop tour

Summary



- desktop tour,
- applications

Summary



- desktop tour,
- applications,
- Linux history, user freedom

Summary



- desktop tour,
- applications,
- Linux history, user freedom,
- distributions, installation

Summary



- desktop tour,
- applications,
- Linux history, user freedom,
- distributions, installation,
- DE ← WM ← CLI; login manager; Vim

Summary



- desktop tour,
- applications,
- Linux history, user freedom,
- distributions, installation,
- DE ← WM ← CLI; login manager; Vim,
- educational channels

Summary



- desktop tour,
- applications,
- Linux history, user freedom,
- distributions, installation,
- DE ← WM ← CLI; login manager; Vim,
- educational channels,
- ricing: conky, polybar, compositor

Summary



- desktop tour,
- applications,
- Linux history, user freedom,
- distributions, installation,
- DE ← WM ← CLI; login manager; Vim,
- educational channels,
- ricing: conky, polybar, compositor,
- Linux phones.

Are you ready to own your computer



Feel free to share your

- adventure (how you liked Linux, new softwares/channels found),
- constructive suggestions (what else would have helped you)!