

Table of Contents

How Do GUI Systems Work In Linux?	2
High Level Overview	2
X11	2
Display Managers	3
Desktop Environments	3

Please note: This is one of my earlier posts from my blog (Jan 2019), and some stuff is missing from it, but it's here as a reference. I also need to break apart the long sections.

How Do GUI Systems Work In Linux?

A common point of confusion on GNU/Linux systems is the display manager vs desktop environment, and what they both do, and how they all fit in with xorg or Wayland.

High Level Overview

When the computer starts up, GNU/Linux only ships with a terminal (tty1-tty6). On top of the terminal, we run a graphics server. Historically, this has been Xorg X11, which came out some ~30 years ago. Newer systems occasionally use Wayland, but support is still lacking for many configurations (such as NVidia cards). Now, X11 just supports “Screens”, mice and keyboard inputs and outputs. Mind you, what X11 considers a “screen” is not just a monitor. I'll go more into this in another post. Most importantly, X11 lets us have graphics, and not just text in a terminal.

Once X11 starts, all we have is an empty screen with a (blank) cursor. We need what we call a Display Manager to actually manage the display. The display manager (DM) starts with your computer (through systemd) and is what pops up the “login” screen, handles which Desktop Environment to load, session management and on some systems, locking the system.

The desktop environment itself is what we see most frequently. It handles all the applications that are running, maximizing, minimizing, stacking windows and the works. It handles the ‘menu’ or task bar, launching applications and essentially is what we think of as the ‘gui’ of the system.

X11

X, or commonly known as Xorg is the display server used in most GNU/Linux systems. It's rather similar to VNC. It makes a local “desktop” and connects to it. X11 is called such as it is the 11th major revision of the X system.

X has support for “screens”, “graphics cards”, mice and keyboards. An X ‘screen’ is not simply a monitor, as if you have 2 or 3 ‘screens’, you can't actually move stuff around! They are independent sessions. (I'll go into some more detail in another post). To get around this limitation, we just make a single screen, with a resolution that spans all of your monitors. (Currently, this is done through xrandr in most computers, previously, Xinerama was pretty common).

If you simply start xorg by itself, you'll just get an empty screen (checker pattern or black) with a crosspoint as a cursor. This is because there isn't really anything running on that server. You can actually use the computer this way, manually launching programs from the terminal, it's not very common (a

common use of this is digital signage, where you don't want any excess bloat – just the one application that needs to run in fullscreen) as you can't resize or maximize programs. Instead, we use a layer of abstraction, the desktop manager.

Display Managers

Technically, you don't need a display manager to run a system. You can directly launch your desktop environment and that would work, but it would essentially “automatically login” and it would be a pain to have 2 users logged on and such (and lock screen in some DEs). This is where the desktop manager comes in. The DM is the only program to get launched with Xorg. It's what prompts you to 'login' into a system.

Display managers handle multiple sessions, such as having 2 different desktop environments running (eg: GNOME on tty1, KDE on tty2 and i3 on tty3) together, or, multiple users or instances running at the same time. They also take care of launching and exiting the desktop environments.

Display managers have little functionality to them most of the time. Just manage sessions through logins and logouts. This is usually handled through PAM (pluggable authentication modules). What that does in short, is whenever someone needs to login (or call sudo, or change settings), it simply calls PAM, which decides whether to ask for a password, or fingerprint and such.

Common display managers include:

- GDM (Gnome Display Manager) – Default on GNOME
- SDDM (Simple Desktop Display Manager) – Default on KDE Plasma
- LXDM (Lightweight X11 Display Manager) – Default on LXDE

Actually, you can use any combination of display manager and desktop environment. Personally, I currently use GDM with KDE on my laptop, since GDM handles PAM authentication for the fingerprint reader well, with KDE as my DE of choice. On my desktop, I use LXDM with KDE (or XFCE) due to the lightweight load.

You should also note that on most desktop environments, locking is handled through the DE, not the DM. On KDE, if I do ctrl+alt+L (lock hotkey), you're prompted with a login wizard that looks like sddm. (Note that PAM is handled a bit differently: it uses system auth in this case instead of the login auth)

Desktop Environments

The desktop environment is the most (easily) visible component of the GUI. It handles the taskbar, menus, maximizing and minimizing applications, stacking applications (things on top of each other), desktop icons & wallpaper, launching most programs, WiFi connections, volumes, clipboards and such. It handles visual aspects as well, including icons, system colors and themes used in applications and other GTK/QT settings.

A lot of essential parts of the system ship with a desktop environment. Every (full) desktop environment ships with its own suite of applications for file browsing, settings, web browsing. As such, the same computer can look and behave vastly different among various computers. This is also due to desktop environments being based on either the GTK+ or QT toolkits.

For example, some common desktop environments:

- GNOME (GTK3)
- KDE (QT5)
- XFCE (GTK2/3)

Desktop environments connect with a lot of other system services to perform functions, in a way that is easy for the user to interact with. For example, the volume sliders on most DE's actually just send commands to PulseAudio, the network manager is usually handled by NetworkManager, and so on.

Essentially, most of what you interact with (outside of applications) is dependent on the DE you choose. Everything from settings, to file managers, to lockscreens, to menus, to icons are determined by the DE. SO MUCH of the look, feel, and functionality can change based on the DE. There is so much to cover that I will probably end up creating another post just on this.

Now even DE's have a core, which actually handles the applications. There are 2 main types: stacking and tiling. Stacking is what most users are used to. Its used on Windows, MacOS and most environments. Its when open applications 'stack' on top of each other. If I open Firefox, then Thunderbird, Thunderbird will appear 'on top' of Firefox, and the keyboard will interact w/Thunderbird. If you click on Firefox, the focus switches, and FF becomes 'on top'. Meanwhile, in tiling desktops (like i3), opening 2 apps opens them side by site, usually following a tree layout.

Common desktop environments include:

- GNOME
- KDE (now called Plasma)
- LXDE (now LXQT is preferred)
- XFCE
- i3 (or Sway for Wayland)
- MATE (like Gnome2)
- Cinnamon

From:

<https://wiki.tonytascioglu.com/> - **Tony Tascioglu Wiki**

Permanent link:

https://wiki.tonytascioglu.com/articles/linux_gui_systems_overview

Last update: **2022-04-12 18:57**

