

# **Configuración inicial de i3wm en Arch Linux**

Javier Niño Sánchez

16 de abril de 2026

Guía de referencia y recopilación de métodos para conseguir un sistema inicial razonable en Arch Linux, partiendo desde la TTY hasta un escritorio minimalista con i3wm (Xorg) o sway (Wayland). Incluye ejemplos de configuración, diagramas conceptuales y buenas prácticas.

# Índice general

<b>1. Introducción</b>	<b>6</b>
1.1. Objetivo	6
1.2. Motivación	6
1.3. Convenciones	6
<b>2. Preparación desde la TTY</b>	<b>8</b>
2.0.1. ¿Qué es un TTY?	8
2.1. Instalación mínima	8
2.1.1. Software para i3wm y sway	9
2.1.2. Software para i3wm	9
2.2. Gestor de paquetes y reflector	11
<b>3. Stack de red</b>	<b>12</b>
3.1. iwctl (iwd)	12
3.1.1. Introducción a iwd	12
3.1.2. Configuración de eduroam	13
3.2. NetworkManager (alternativa)	13
3.3. Configuración de DNS / resolv.conf	14
<b>4. Stack gráfico: i3wm (Xorg)</b>	<b>16</b>
4.1. Diferencias conceptuales: Xorg vs Wayland	16
4.1.1. Qué son los servidores gráficos	16
4.1.2. ¿Qué servidor gráfico es recomendable utilizar?	17
4.2. i3wm (Xorg)	17
4.2.1. Estructura de <code>/.config/i3/config</code>	17
4.2.2. Compositores y picom	18
4.2.3. Rofi, el good for all de i3wm	19
4.3. Recursos gráficos compartidos	21
4.3.1. Fuentes	21
4.3.2. Iconos	21
4.3.3. Temas GTK	21
4.3.4. Temas Qt	22
4.3.5. Cursor	22
<b>5. Polkit, permisos y pulseaudio/pipewire</b>	<b>23</b>
5.1. polkit: por qué y ejemplos de reglas simples	23
5.1.1. Agente de polkit	23
5.1.2. Reglas de autorización	23
5.2. Audio: PipeWire como backend moderno	24
5.2.1. Instalación básica	24
5.2.2. Integración con herramientas de PulseAudio	25
5.2.3. Bluetooth con PipeWire y aplicaciones recomendadas	25
<b>6. Ricing y estética</b>	<b>26</b>
6.1. <code>.Xresources</code> y <code>notify-send</code>	26
6.1.1. Instalación de <code>notify-send</code> y <code>dunst</code>	26

6.1.2. startx y como se lanzan los servidores X11 . . . . .	26
6.2. Wallpaper, bar (i3bar) y launchers (rofi) . . . . .	27
<b>7. Solución de problemas y checklist</b>	<b>30</b>
7.1. Logs relevantes (journalctl) . . . . .	30
7.1.1. journalctl . . . . .	30
7.1.2. Xorg.0.log (solo X11) . . . . .	30
7.2. Problemas frecuentes . . . . .	31
7.2.1. Como puedo establecer el teclado de mi idioma a español . . .	31
7.2.2. Como habilitar el tap para hacer click en el touchpad . . . . .	31
7.2.3. Como hacer hot-plugging de múltiples monitores en X11 . . . .	31
<b>A. Configuración i3wm en portátil</b>	<b>37</b>
<b>B. Configuración rofi de ejemplo</b>	<b>48</b>

# Índice de figuras

3.1. Flujo desde el servidor DHCP hasta DNS . . . . .	14
7.1. Flujo desde la regla udev hasta el hotplug de monitores . . . . .	31

# 1. Introducción

## 1.1. Objetivo

Este manual pretende recopilar el proceso de configuración de un sistema basado en GNU/Linux[1] que utiliza gestores de ventanas manuales como i3[2] y sway[3]. Se ofrecen soluciones a las posibles necesidades de los usuarios de dichos gestores de ventana. Este manual existe con el objetivo de centralizar la documentación para su futura consulta.

Aunque aparece sway en la documentación, la versión actual del manual solo está diseñada para i3wm. En un futuro se incluirá la versión para sway. Por suerte, como sway ha sido diseñado como un drop-in debería de resultar sencillo al lector adaptar buena parte del manual para sway.

La distribución utilizada en el manual es Arch Linux, esto se debe a su extensa documentación, instalación base minimalista y el software disponible a través del gestor de paquetes[4]<sup>1</sup> pacman[5].

Se busca demostrar que no es necesario tener un entorno de escritorio para tener un sistema perfectamente funcional, y que, por lo tanto, no es necesario depender exclusivamente en un suite de software que en un futuro puede cambiar a un estado no deseado.

## 1.2. Motivación

Este manual pretende compartir conocimientos relacionados con el funcionamiento de Linux, así como ofrecer una guía básica que pueda ser utilizada para empezar a configurar un sistema GNU/Linux.

Esta guía pretende ser un documento en el que se pueda ya no solo aprender a configurar i3wm o sway, sino que sea un resumen del funcionamiento del sistema operativo para futuro estudio.

## 1.3. Convenciones

En este manual, se establecerá `$XDG_CONFIG_HOME` en `$HOME/.config`, de forma que la mayoría de la configuración se realizará en este directorio. A su vez, se recomienda tener un directorio auxiliar (`$HOME/.i3`) para guardar posibles configuraciones extra y scripts utilizados para i3wm.

---

<sup>1</sup>Un gestor de paquetes es software utilizado, generalmente en distribuciones para facilitar la instalación, gestión y eliminación de software de manera controlada

Por regla general se utilizará BASH (Bourne Again Shell) para los comandos utilizados, así como lenguaje de scripting. Para los directorios, se usará como base \$HOME, usando como locale es\_ES-utf8

Se asume que el usuario no ha instalado ningún entorno de escritorio, pero este manual se puede seguir perfectamente aún si se ha instalado algún entorno.

El nombre de los paquetes a instalar vendrá monoespaciado, a su vez, si los paquetes deben de ser instalados desde la AUR se usará un superíndice como el que sigue paquete<sup>AUR</sup>

## 2. Preparación desde la TTY

### 2.0.1. ¿Qué es un TTY?

Un TTY[6]<sup>1</sup> en GNU/Linux supone un emulador de terminal<sup>2</sup> en el cual se pueden ejecutar comandos para interactuar con el kernel y los recursos del hardware. Usualmente, en sistemas Linux habrá 7 TTYs virtuales, accesibles mediante `Ctrl`+`Alt`+`F1` hasta `Ctrl`+`Alt`+`F6`.

Por regla general, las distribuciones usarán la primera TTY para el login y la última para el entorno gráfico, pero esto depende de cada distribución y de como se gestione la interfaz gráfica.

Cada TTY tiene sus propios streams de entrada y salida, por lo que permite usar múltiples terminales de manera independiente. En nuestro caso, la TTY actuará como terminal principal para la instalación inicial, no nos preocuparemos porque la interfaz gráfica esté en la última TTY o en la primera, por regla general, el greeter se encargará de gestionar estos asuntos.

### 2.1. Instalación mínima

Hay que descargar múltiples paquetes para que el sistema pueda funcionar correctamente, nos basaremos en la guía de instalación de Arch Linux[7], que incluye los siguientes paquetes:

- `amd-ucode` o `intel-ucode`: Microcode para la CPU del sistema. **Instalar solo uno.**
- `linux-firmware`, `sof-firmware`: Instalacion del firmware necesario, `linux-firmware` es firmware general que podría ser necesario para `mkinitcpio` y otras utilidades del sistema, `sof-firmware` es firmware dedicado para audio. **Instalar ambos.**
- `iw` o `NetworkManager`: Gestores del stack de red desde el espacio de usuario. Se pueden utilizar software como `wpa_supplicant`, pero para en el manual se usará `iw` o `NetworkManager`. Se recomienda **instalar solo uno**, puede haber conflictos si ambos están funcionando en el sistema al mismo tiempo. Se priorizará `iw`, ya que no hay buenos argumentos para utilizar `NetworkManager` (que puede usar `iw` como backend) si solo se quiere gestionar conexiones Wi-Fi.
- `nano`, `vim`, `nvim` y `emacs`: Editores de texto destacados. **Se recomienda instalar nano por lo menos.**
- `man`: Documentación para sistemas POSIX.

---

<sup>1</sup>TeleTypewriter

<sup>2</sup>Se entiende por emulador de terminal todo software que intente simular el comportamiento de las terminales originales que tenían los ordenadores antiguamente



Aunque no es mencionado directamente en la guía, se recomienda instalar un polkit[8] para la gestión de sesiones. Observando las opciones resulta evidente que habrá que tener cuidado con la elección que hagamos. Nuestro objetivo es encontrar un polkit ligero y que sea compatible tanto con X11 y sway, trayendo consigo el menor número de dependencias posible.

En base a estos criterios (aunque se recomienda experimentar con otros polkit) se usará polkit-gnome.

También hay que instalar software adicional dependiendo del gestor de ventanas que utilicemos. A continuación se detallará el software que se necesita o se recomienda instalar.

### 2.1.1. Software para i3wm y sway

- Alacritty: Emulador de terminal<sup>3</sup> recomendado.
- htop, nvtop, btop: TUIs<sup>4</sup> para gestión de recursos. htop es sobre todo para procesador, nvtop es para GPUs y btop es para el sistema en general.
- brightnessctl: CLI para la gestión del brillo de las pantallas. Muy útil para scripts.
- impala: TUI para gestión de conexiones Wi-Fi.
- bluetoothtui: TUI para gestión de dispositivos bluetooth.
- ranger: TUI file manager.
- yay: Gestor de paquetes de la AUR.
- fzf: Fuzzy finder para archivos del sistema. Extremadamente potente.

En base a esta lista, es fácil percatarse de que la mayoría de aplicaciones mencionadas no dependen del servidor gráfico que el usuario haya iniciado en su sesión actual. Esto se debe a que, para esta sección, se ha buscado maximizar la compatibilidad de los programas con los sistemas que se basan en i3wm (X11) y sway (wayland).

### 2.1.2. Software para i3wm

- i3: Paquete principal de i3wm. Incluye los paquetes i3-wm (window manager), i3blocks i3status (programas para escribir status en i3bar mediante stdout) y i3lock (screen locker).
- feh: Programa simple para visualizar imágenes, y establecer fondos de pantalla. Si se quiere una aplicación que se encargue exclusivamente de establecer fondos de pantalla, se puede utilizar nitrogen.
- rofi: Posiblemente de los mejores programas que se han creado para X11, es un program launcher que permite la creación de modos. Permite hacer de todo, en este manual se utilizará junto rofimoji (listado de emojis para poder utilizarlos de manera conveniente), rofi-calc (calculadora rápida). Pero también se puede usar como un power menu o como un gestor de portapapeles.

---

<sup>3</sup>Gestores de ventana como i3wm ya incluyen su propia terminal, pero es preferible usar otras

<sup>4</sup>Terminal User Interface, un programa cuya interfaz gráfica se renderiza en el terminal, normalmente usando librerías como ncurses.

- `maim`: Para capturas de pantalla. Se puede utilizar `scrot`, pero este último es bastante anticuado
- `xorg-xrandr`: Para gestión de resoluciones, tasas de refresco y posicionamiento de las pantallas.
- **Opcionales:**
  - `polybar`: Programa para reemplazar `i3bar` en el caso de que se quiera configurar a mayor profundidad el status-line.
  - `clipmenu`: Programa para gestión de historial del portapapeles.
  - `rofi-bluetooth`: Modo de `rofi` para gestión de dispositivos bluetooth conectados.

En esta sección no se han incluido los programas para el stack de red, esto se debe a que en el siguiente capítulo se tratará la configuración del stack de red según los paquetes que se utilicen.

## 2.2. Gestor de paquetes y reflector

El paquete reflector [9] es un script de python que se encarga de actualizar la lista de espejos utilizada para instalar paquetes usando pacman. Dicho paquete incluye un .service y un .timer que se pueden configurar. Reflector permite configurar la generación de la lista de espejos, modificando el archivo de configuración que se encuentra en /etc/xdg/reflector/reflector.conf.

Un ejemplo de configuración, en el que se ordenan por velocidad y se eligen solo los 10 mejores sería:

```
1  # Reflector configuration file for the systemd service.
2  #
3  # Empty lines and lines beginning with "#" are ignored. All other lines should
4  # contain valid reflector command-line arguments. The lines are parsed with
5  # Python's shlex modules so standard shell syntax should work. All arguments are
6  # collected into a single argument list.
7  #
8  # See "reflector --help" for details.
9
10 # Recommended Options
11
12 # Set the output path where the mirrorlist will be saved (--save).
13 --save /etc/pacman.d/mirrorlist
14
15 # Select the transfer protocol (--protocol).
16 --protocol https
17
18 # Select the country (--country).
19 # Consult the list of available countries with "reflector --list-countries" and
20 # select the countries nearest to you or the ones that you trust. For example:
21 # --country France,Germany
22 --country ES,PT,GB,DE,IT
23
24 # Use only the most recently synchronized mirrors (--latest).
25 --latest 15
26
27 # Sort the mirrors by synchronization time (--sort).
28 --sort rate
```

## 3. Stack de red

### 3.1. iwctl (iwd)

#### 3.1.1. Introducción a iwd

`iwd`<sup>[10]</sup> es un gestor de conexiones inalámbricas desarrollado por intel como sustituto de `wpa_supplicant`. El objetivo es ser más eficiente con los recursos a la hora de gestionar dichas conexiones.

`wpa_supplicant` es un programa para conexiones WPA, WPA2, WPA3<sup>1</sup> que actúa como suplicante, puede ser utilizado por programas como NetworkManager como backend al igual que `iwd`. La principal diferencia es que `iwd` intenta aprovechar lo máximo posible la funcionalidad ofrecida por el kernel, mientras mantiene reducido el número de dependencias.

Se puede usar como standalone (que es como se utilizará en esta guía), también se puede usar como backend para NetworkManager, systemd-networkd y otros.

Hay que tener en cuenta que este guión está pensado para utilizarse junto a systemd, así que la configuración que se recomienda utilizar estará basada en el uso de systemd-resolved para la configuración del DNS (más información en el punto 3.3)

Aunque en la Wiki de Arch se menciona, es recomendable instalar algún front-end como interfaz gráfica, en este caso se recomienda `impala`, una TUI moderna que funciona muy bien en cualquier entorno gráfico.

Aunque se use `impala` (o cualquier front-end gráfico que se quiera), es recomendable saber manejarse con `iwd` directamente. Por un lado se pueden ejecutar comandos mediante el modo interactivo:

```
1 $ iwctl #ejecutar modo interactivo
2 $ [iwd] device list #ver las interfaces del sistema, para el ejemplo se asume que
   → wlan0 es la interfaz de Wi-Fi
3 $ [iwd] station wlan0 scan #scanear redes Wi-Fi
4 $ [iwd] station wlan0 get-networks #devuelve una lista (que se puede seguir
   → actualizando si sigue scaneando) con las redes disponibles
5 $ [iwd] station wlan0 connect SSID #SSID es el nombre de la Wi-Fi, pedirá una
   → contraseña
6 $ [iwd] station wlan0 disconnect #desconectarse de la red
7 $ [iwd] station wlan0 show #muestra información de la interfaz
8 $ [iwd] help #Ayuda sobre comando extremadamente útil
```

Se puede hacer la conexión en un solo comando, pero es recomendable usar el modo interactivo si no se está usando un front-end.

---

<sup>1</sup>En otras palabras, para conexiones Wi-Fi

### 3.1.2. Configuración de eduroam

La UGR ofrece un script de python que hace el trabajo de configurar eduroam automáticamente, sin necesidad de que el estudiante tenga que configurarlo manualmente. El principal problema de esta configuración es que depende de que se esté utilizando NetworkManager, ya que el script asume que el sistema está basado en algún entorno de escritorio (que por defecto suelen traer NetworkManager en la mayoría de entornos populares)

Debido a esto, el script fallará cuando se intente lanzar en sistemas basados en iwd, ya que busca la interfaz de NetworkManager. Una de las posibles soluciones es la configuración manual, la cual escribiremos en /var/lib/iwd, que es donde iwd espera encontrar las configuraciones de red. Para ello, escribiremos en el archivo /var/lib/iwd/eduroam.8021x

```

1  [Security]
2  EAP-Method=PEAP
3  EAP-Identity=anonymous2022@ugr.es
4  EAP-PEAP-CA Cert=/var/lib/iwd/CA-UGR-eduroam.pem
5  EAP-PEAP-ServerDomainMask=*.ugr.es
6  EAP-PEAP-Phase2-Method=MSCHAPV2
7  EAP-PEAP-Phase2-Identity=estudianteejemplo@correo.ugr.es
8  EAP-PEAP-Phase2-Password=passwordejemplo
9
10
11 [Settings]
12 AutoConnect=true

```

Asegúrate de haber instalado el certificado de la UGR y que esté en /var/lib/iwd, es probable que tengas que convertir el certificado .cert a .pem como en el ejemplo. Para ello, puedes ejecutar el siguiente comando.

```

1  /var/lib/iwd $ openssl x509 -in CA-UGR-eduroam.crt -outform PEM -out
    ↪ CA-UGR-eduroam.pem

```

## 3.2. NetworkManager (alternativa)

La configuración de NetworkManager es bastante directa en comparación<sup>2</sup>, una vez instalado el paquete bastaría con hacer lo siguiente (antes de lanzar el servicio):

```

1  $ sudo systemctl enable --now systemd-resolved #habilitar systemd-resolved
2  $ sudo ln -s /run/systemd/resolve/stub-resolv.conf /etc/resolv.conf #enlazar el
    ↪ stub a resolv.conf

```

Tras ello, antes de iniciar NetworkManager hay que actualizar su configuración en /etc/NetworkManager/NetworkManager.conf<sup>3</sup> para incluir la siguiente línea:

<sup>2</sup>uno de sus puntos a favor en comparación con iwd

<sup>3</sup>Se puede crear un archivo en /etc/NetworkManager/conf.d/ en caso de que el usuario quiera separar la configuración de distintos aspectos de NetworkManager.

```

1 [main]
2 dns=systemd-resolved

```

Tras ello, sería cuestión de lanzar NetworkManager:

```

1 $ sudo systemctl enable --now NetworkManager #habilitar NetworkManager

```

Si por algún motivo NetworkManager ya estuviera funcionando, entonces es tan simple como reiniciar los servicios:

```

1 $ sudo systemctl restart systemd-resolved
2 $ sudo systemctl restart NetworkManager

```

Y para comprobar que está funcionando:

```

1 $ resolvectl status #resolvectl es el cli de systemd-resolved
2 $ ls -la /etc/resolv.conf #debería de devolver el enlace creado anteriormente

```

### 3.3. Configuración de DNS / resolv.conf

Hay muchas formas de configurar `/etc/resolv.conf` y muchos argumentos para usar un método u otro. Entre los métodos se incluye la configuración por NetworkManager, o mediante el uso de programas específicos basados en DHCP como `dhclient` (esta clase de solución es la más coherente con la filosofía UNIX<sup>4</sup>).

Si el lector ha optado por utilizar NetworkManager, entonces no necesitará preocuparse de popular `/etc/resolv.conf`, puesto que en su configuración por defecto este gestor de conexiones ya edita dicho archivo (normalmente a través de `systemd-resolved`). En ese caso, el flujo sería:

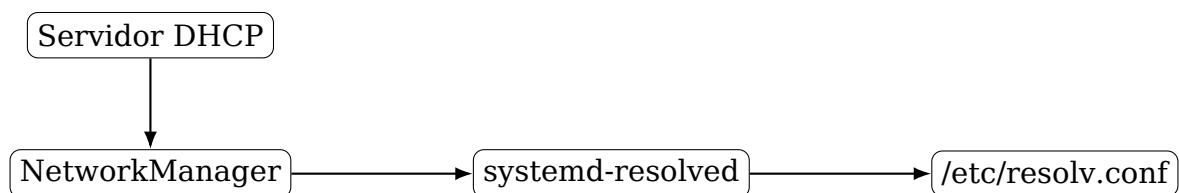


Figura 3.1.: Flujo desde el servidor DHCP hasta DNS

Aunque lo que queda de sección concierne a aquellos usuarios que hayan optado por usar `iwid`, se ha dejado como una sección a parte en el caso en el que los usuarios de NetworkManager quieran tener varios métodos para completar `/etc/resolv.conf`, en cuyo caso se aconseja instalar `openresolv` para la sincronización de escrituras sobre el archivo.

En este caso aprovecharemos que `iwid` tiene la capacidad de interactuar con el servidor DHCP y usaremos `systemd-resolved` para acabar con un flujo parecido al

<sup>4</sup>“Do one thing and do it well”, el software debería de ser minimalista y modular.

que tiene NetworkManager.

Para ello es necesario habilitar systemd-resolved, esto creará un stub que escuchará en el puerto 53 (puerto para DNS), de esta manera cuando se quiera consultar un dominio, resolv.conf apuntará a dicho puerto y systemd-resolved hará el resto del trabajo. En una terminal:

```
1 $ sudo systemctl enable --now systemd-resolved #habilitar systemd-resolved
2 $ sudo ln -s /run/systemd/resolve/stub-resolv.conf /etc/resolv.conf #enlazar el
   ↪ stub a resolv.conf
```

Si observamos /etc/resolv.conf, debería de aparecer algo como lo que sigue:

```
1 # This is /run/systemd/resolve/stub-resolv.conf managed by man:systemd-resolved(8).
2 # Do not edit.
3 #
4 # This file might be symlinked as /etc/resolv.conf. If you're looking at
5 # /etc/resolv.conf and seeing this text, you have followed the symlink.
6 #
7 # This is a dynamic resolv.conf file for connecting local clients to the
8 # internal DNS stub resolver of systemd-resolved. This file lists all
9 # configured search domains.
10 #
11 # Run "resolvectl status" to see details about the uplink DNS servers
12 # currently in use.
13 #
14 # Third party programs should typically not access this file directly, but only
15 # through the symlink at /etc/resolv.conf. To manage man:resolv.conf(5) in a
16 # different way, replace this symlink by a static file or a different symlink.
17 #
18 # See man:systemd-resolved.service(8) for details about the supported modes of
19 # operation for /etc/resolv.conf.
20
21 nameserver 127.0.0.53
22 options edns0 trust-ad
23 search .
```

Lo único que faltaría sería configurar iwd para que utilice systemd-resolved, en /etc/iwd/main.conf hay que escribir:

```
1 [General]
2 EnableNetworkConfiguration=true
3
4 [Network]
5 NameResolvingService=systemd
```

## 4. Stack gráfico: i3wm (Xorg)

### 4.1. Diferencias conceptuales: Xorg vs Wayland

El objetivo de este manual es ofrecer una guía para configurar gestores de ventana como i3wm o sway, pero es de interés entender como funcionan los servidores gráficos que permiten el funcionamiento de las interfaces de usuario.

#### 4.1.1. Qué son los servidores gráficos

Un servidor gráfico es un programa que se encarga de gestionar recursos limitados relacionados con interfaces gráficas. En otras palabras, esta clase de programas son los responsables de la comunicación entre programas gráficos mientras gestiona los recursos<sup>1</sup> que a cada programa se le permite ocupar.

En la actualidad, esta clase de programas son muy complejos, pero en Linux hay principalmente dos: X11 y Wayland. X11[11] es la versión número 11 del protocolo X, la cual lleva siendo la versión principal por más de 40 años. Este protocolo tiene el problema de que su estructura está basada en entornos gráficos más simples, por lo que tiene peor soporte de sistemas multi monitor y/o con distintas resoluciones. A su vez, el protocolo tiene pocas comprobaciones de seguridad sobre la información a la que pueden acceder a los distintos procesos, lo cual, desde un punto de vista de scripting, es muy útil, pero a su vez reduce mucho la seguridad del sistema. El lado bueno es la estabilidad respecto al hardware, suele usarse como fallback en muchos casos y tiende a funcionar mejor con las gráficas de Nvidia.

Por otro lado está wayland[12], el cual es un proyecto diseñado para solucionar los problemas que presentaba X11 que tiene ya 17 años de desarrollo. Este protocolo carece de los problemas clásicos de X11,<sup>2</sup> en cambio, entre sus principales problemas se tiene que el entorno de desarrollo sigue siendo inmaduro,<sup>3</sup> tiende a tener mayor inestabilidad (tanto por software como hardware).<sup>4</sup>

---

<sup>1</sup>Espacio en la pantalla, superposición de programas, configuración de pantallas y framebuffers, DPI...

<sup>2</sup>el soporte multi monitor, sobre todo al hacer hot-plugging, es muy bueno. También mejora muchísimo la seguridad del sistema

<sup>3</sup>cabe recalcar que con cada año va mejorando, hay gran aceptación por parte de las distribuciones, pero sigue teniendo problemas en cuanto a software para automatización y scripting.

<sup>4</sup>Este es el principal problema/crítica que se puede hacer al proyecto, que se realizó como un "borrón y cuenta nueva", lo que implica que a lo largo de los 17 años se han encontrado con multitud de edge-cases que ya estaban solucionados en X11.



### 4.1.2. ¿Qué servidor gráfico es recomendable utilizar?

Este apartado está dedicado a aquellos lectores que consulten este manual sin tener una intención clara de qué servidor gráfico utilizar. Debido a que ya se han expuesto a grandes rasgos los beneficios e inconvenientes de cada modelo quedaría hacer una síntesis de los perfiles de usuario que podrían interesarse por un servidor u otro.

Se recomienda instalar X11 (y por lo tanto i3wm) al usuario si prioriza el control sobre su sistema, la compatibilidad con hardware (en caso de que wayland le de problemas), la extensa documentación que hay sobre el software y si se conforma con un sistema menos seguro.

Se recomienda instalar wayland (y por lo tanto sway) al usuario si prioriza la seguridad del sistema, un software que en la actualidad recibe mucho mantenimiento y mejoras, la buena documentación que hay sobre el software y si se conforma con un menor control del sistema, alguna incompatibilidad con el software (mayormente solucionado con xwayland) y no tiene problemas con su hardware.

## 4.2. i3wm (Xorg)

i3wm es un gestor de ventanas en mosaico tiling window manager<sup>5</sup> para X11. La filosofía es la simplicidad, la configuración principal se hace en un único fichero de texto plano y el comportamiento del sistema está controlado por el teclado.

En secciones anteriores se ha cubierto el software a instalar para lograr una instalación mínima del sistema, así que en estas secciones se explicará la estructura básica de configuración y el software más relevante.

### 4.2.1. Estructura de `/.config/i3/config`

Como se ha comentado anteriormente, i3wm cuando inicia busca el archivo `/.config/i3/config` para obtener la configuración realizada por el usuario. Aunque se puede usar la keyword **include** para utilizar múltiples archivos.

Al iniciar i3wm por primera vez este archivo no existirá, por lo que i3wm invocará una ventana preguntando al usuario si quiere que genere la configuración por defecto. Para los usuarios con poca experiencia se les recomienda empezar con la configuración por defecto.

Dicho esto, se recomienda mirar la documentación para entender el funcionamiento del fichero de configuración[13]. Aún así, en los apéndices se incluyen ejemplos de configuración para demostrar el potencial de i3wm (debido a su longitud, no se incluyen aquí).

---

<sup>5</sup>Es decir, basado en árboles de estado, siendo el usuario el que controla la distribución de las ventanas manualmente.

### 4.2.2. Compositores y picom

Este apartado es opcional, no es necesario tener un compositor para que i3wm funcione. Pero es útil para añadir animaciones y para arreglar el posible screen tearing causado por la falta de uno.

Un compositor[14] es software que contiene buffers de las ventanas gráficas y combina efectos visuales para generar frames. De esta manera, se puede añadir animaciones, solucionar problemas como screen tearing, aplicar transparencia a ventanas, entre otros.

Entre todos los compositors que hay disponibles, se recomienda utilizar picom, ya que es uno de lo más actualizados y eficientes[15]. El motivo por el que hay que instalar un compositor standalone surge por la filosofía minimalista de i3wm<sup>6</sup>. Este gestor de ventanas no asume nada sobre la configuración del sistema, por lo que el usuario tiene que tomar esta clase de decisiones por su cuenta.

Picom tiene buena documentación, se pueden encontrar comentarios explicando que hace cada cosa en el archivo de configuración que se puede encontrar en /usr/share/doc/picom/picom/picom.conf.example. También se pueden establecer reglas sobre opacidad, sombreado de ventanas, entre otros. Es recomendable copiar el este documento a /.config/picom.conf. Una de las posibles modificaciones a la configuración encontrada en el sistema

```

1 #####
2 # Transparency / Opacity #
3 #####
4
5 # Opacity of window titlebars and borders.
6 #
7 # Range: 0.1 - 1.0
8 # Default: 1.0 (disabled)
9 # Deshabilitamos opacidad general
10 frame-opacity = 1.0;
11
12 # añadimos opacidad por ventana
13 # estructura:
14 # "<% opacidad>:<atributo de ventana> = <valor atributo>"
15 opacity-rule = [
16     "95:class_g = 'Spotify'", #95% opacidad para Spotify
17 ];
18
19 #####
20 # Blur #
21 #####
22
23 # Parameters for background blurring, see BLUR section in the man page for more
24 ↪ information.
25 blur-method = "dual_kawase"
26 blur-strength = 3
27 # Specify the blur convolution kernel, with the following format:

```

<sup>6</sup>Sway, aunque diseñado como un drop-in de i3wm para Wayland, tiene integrado un compositor, pero esto tiene que ver más con el funcionamiento de wayland que por la propia filosofía del gestor de ventanas



- `rofimoji`: Integración básica de Unicode para caracteres y emojis. (Recomendable)
- `rofi-emoji`: Un selector de emojis integrado. (Extremadamente recomendable)
- `rofi-power-menu`<sup>AUR</sup>: Permite usar rofi como un menú para apagar, suspender, reiniciar... Especialmente útil si se plantea utilizar rofi para esta clase de tareas. (Recomendable según el caso)
- `rofi-calc`: Permite realizar operaciones y conversiones entre unidades. (Extremadamente recomendable).

También se puede utilizar junto a algunos gestores de portapapeles como `clipmenu`, pero no suelen ser plug-and-play como todos los paquetes que hemos enseñado anteriormente.

Además de ser un desktop application launcher<sup>7</sup>, rofi permite la configuración, tanto de los modos<sup>8</sup> de uso como de la GUI. Por lo tanto, a continuación se presentan dos posibles configuraciones, una general, y otra específica de tema.

Por un lado, la configuración general existe en el archivo `/.config/rofi/config.rasi`:

```

1 configuration {
2     modi: "run,drun>window,emoji,calc";
3     font: "Iosevka 18";
4     fixed-num-lines: true;
5     show-icons: true;
6     display-run: "Applications:";
7     display-window: "Windows:";
8     timeout {
9         action: "kb-cancel";
10        delay: 0;
11    }
12 }
13 @theme "~/.config/rofi/nord.rasi"
14
15 listview {
16     lines: 8;
17     columns: 2;
18 }
19
20 window{
21     width: 45%;
22 }
23
```

Por otro lado, y como se presenta en la configuración general, los temas pueden residir en cualquier parte del sistema, pero por convención se recomienda utilizar el directorio `/.config/rofi/` o `/.config/rofi/themes`. En este caso, la configuración se ha añadido en uno de los apéndices.

En este ejemplo se ha usado el tema de nord, adaptado por undiabler. Pero se recomienda al usuario a crear el suyo propio a partir de uno existente con la ayuda de la documentación de rofi[18].

<sup>7</sup>En este caso, una subclase caracterizada por una interfaz minimalista, de control por teclado y diseñado para lanzar aplicaciones o ejecutar acciones rápidamente sin navegación gráfica tradicional.

<sup>8</sup>En términos de rofi, los modi

### 4.3. Recursos gráficos compartidos

Esta sección aún está en desarrollo, por suerte la documentación de arch es extensa, así que se recomienda buscar información ahí.

Independientemente de si se usa i3wm o sway, existen recursos gráficos que se comparten entre sí, como vienen a ser las fuentes, los iconos y los temas GTK/Qt. En esta sección se busca resumir la configuración de estos recursos para que el usuario pueda empezar a explorar posibles opciones.

#### 4.3.1. Fuentes

Se recomienda la instalación de fuentes como `nerd-fonts`, ya que tienden a incluir muchos iconos en sus fuentes y hay una amplia selección de tipografías. Tanto monoespaciadas como fuentes con características especiales. Tanto i3wm como sway las manejan bien, por lo que no deberían de dar problemas de ningún tipo.

Es altamente recomendable instalar más fuentes, como `ttf-dejavu`, `noto-fonts` y `noto-fonts-emoji`. Para comprobar si se ha instalado una fuente, se puede ejecutar

```
1 $ fc-list | grep -i "JetBrains" #por ejemplo, buscar si JetBrains está instalado
```

En el caso de haber instalado manualmente una fuente, es recomendable actualizar la caché del sistema, esto es posible con:

```
1 $ fc-cache -fv #de esta manera regeneramos la caché del sistema.
```

Para más información, se recomienda consultar la documentación oficial[19].

#### 4.3.2. Iconos

Los gestores de archivos y algunas barras de estado (como waybar o polybar) usan paquetes de iconos GTK. Los más usados son:

- `papirus-icon-theme`: icono plano moderno, amplia cobertura.
- `numix-icon-theme`: alternativa clásica y ligera.
- `hicolor-icon-theme`: tema base requerido por GTK, por lo que debe de estar instalado siempre.

Para más información, se recomienda consultar la documentación oficial[20].

#### 4.3.3. Temas GTK

Las aplicaciones GTK, como son Thunar, gedit, entre otros, leen su tema desde `/.config/gtk-3.0/settings.ini` y, si son programas basados en versiones más antiguas, desde `/.gtkrc-2.0`.

Existen tanto temas oscuros como claros, en este sentido, hay que ir buscando específicamente el paquete que sea desea descargar. Una vez elegidos los temas

correctos, se pueden configurar modificando los archivos mencionados anteriormente, para ello, se dejan a continuación unos ejemplos de configuración:

En Wayland la herramienta `nwg-look` (AUR) proporciona una interfaz gráfica equivalente a `lxappearance` para configurar temas GTK sin necesidad de editar los ficheros a mano. En X11 puedes usar directamente `lxappearance`.

#### 4.3.4. Temas Qt

Las aplicaciones Qt no leen la configuración GTK. Para que sigan el mismo tema se usan los paquetes `qt5ct` y `qt6ct`. Para que sean efectivos los cambios, antes hay que decirle a los programas donde deberían de buscar la configuración de QT, por ello, se añade a `.bashrc` la siguiente variable de entorno:

```
1 export QT_QPA_PLATFORMTHEME=qt5ct
```

Tras ello, es cuestión de ejecutar `qt5ct` y elegir los temas que se deseen utilizar.

#### 4.3.5. Cursor

El tema de cursor es independiente al servidor gráfico que se utilice, al igual que con los temas anteriores. Hay muchos temas de cursores, así que por simpleza se destaca `Adwaita`, que se encuentra incluido en `gnome-themes-extra` y es un cursor popular y ligero.

Para aplicarlo globalmente hay que configurar `/.icons/default/index.theme`, a continuación se propone un ejemplo de configuración<sup>9</sup>:

```
1 # This file is written by LXAppearance. Do not edit.
2 [Icon Theme]
3 Name=Default
4 Comment=Default Cursor Theme
5 Inherits=Vimix-cursors
```

<sup>9</sup>Como se puede observar en el ejemplo, el archivo se creó usando `lxappearance`, que se ha mencionado anteriormente como una opción para configurar los temas GTK

## 5. Polkit, permisos y pulseaudio/pipewire

### 5.1. polkit: por qué y ejemplos de reglas simples

polkit es un toolkit/framework de autorización que permite a procesos sin privilegios ejecutar acciones que normalmente requerirían permisos de superusuario. El ejemplo más básico es apagar el equipo. Si el lector ha estado siguiendo el manual y en algún momento ha reiniciado, se habrá percatado de que es necesario hacerlo como superusuario, mientras que en la mayoría de los entornos de escritorio este problema ya está solucionado (ya que suelen incluir su propio polkit ya configurado).

- Montar y desmontar unidades desde un gestor de archivos (Thunar, ranger).
- Suspender, hibernar o apagar el sistema desde un menú de usuario.
- Gestionar redes con NetworkManager sin ser root.
- Cambiar la hora del sistema o instalar impresoras.

#### 5.1.1. Agente de polkit

polkit se compone de un demonio (polkitd) y un *agente*: una pequeña aplicación gráfica o de terminal que muestra el diálogo de autenticación cuando una acción lo requiere.

En entornos minimalistas el demonio se instala con el sistema pero el agente debe arrancarse manualmente. Para i3wm y sway la opción más sencilla es polkit-gnome, como comentábamos anteriormente.

Es necesario a su vez lanzarlo al inicio de sesión, por lo que se debería de incluir en `/.config/i3/config` la siguiente línea:

```
1 exec --no-startup-id /usr/lib/polkit-gnome/polkit-gnome-authentication-agent-1
```

Otras opciones de agente más ligeras son `lxqt-policykit` o `mate-polkit`. Instala y ejecuta solo uno, ya que tener varios agentes activos simultáneamente puede causar diálogos duplicados. Para comprobar si hay varios polkits corriendo, basta con hacer

```
1 $ ps aux | grep polkit
```

A su vez, asegúrate de que el servicio de systemd polkit esté corriendo (debería de aparecer un proceso llamado polkitd).

#### 5.1.2. Reglas de autorización

Polkit usa reglas escritas en JavaScript para versiones modernas. Las reglas escritas por el usuario van en `/etc/polkit-1/rules.d/`, con extensión `.rules`.

Los ficheros, como con muchos directorios de configuración, se cargan en orden lexicográfico, es decir, los prefijos numéricos controlan la prioridad.

A su vez, las políticas que tienden a generarse automáticamente se pueden encontrar en `/usr/share/polkit-1/actions/`, se recomienda encarecidamente no editar estas políticas a no ser que sea estrictamente necesario, es mejor crear reglas y dejar que polkit haga su trabajo.

El motivo por el que no se entra en detalle es porque no se recomienda a los usuarios tocar las reglas de polkit si no tienen experiencia. Aunque se pueden usar para relajar y fortalecer la seguridad el sistema, es preferible no alejarse mucho del default.

## 5.2. Audio: PipeWire como backend moderno

Pipewire[21] es un servidor de audio y vídeo moderno que supone una mejora respecto a pulseaudio en cuanto a la gestión de recursos y eficiencia. Pipewire busca reemplazar PulseAudio<sup>1</sup> y JACK<sup>2</sup>. A pesar de ello, tiene compatibilidad a nivel de API con estos paquetes, por lo que podemos aprovechar la madurez de las herramientas desarrolladas para pulseaudio con la eficiencia que ofrece pipewire.

### 5.2.1. Instalación básica

Es necesario instalar los siguientes paquetes:

- pipewire: demonio principal para el audio y vídeo.
- pipewire-pulse: capa de compatibilidad con PulseAudio<sup>3</sup>.
- pipewire-alsa: redirige ALSA<sup>4</sup> a Pipewire.
- pipewire-jack: compatibilidad con JACK.
- wireplumber: gestor de sesiones para PipeWire.

Una vez instalado, hay que entender que como parte de la arquitectura pipewire se ejecuta como un servicio de usuario, no de sistema. Esto es más una cuestión de seguridad que otra cosa, se puede cambiar, pero es poco recomendable. Dicho eso, bastaría con asegurarse de que pulseaudio no está instalado o corriendo, y, tras ello, podemos iniciar pipewire:

```
1 $ systemctl --user enable --now pipewire pipewire-pulse wireplumber
```

Tras ello, podemos empezar a utilizar las herramientas de PulseAudio en un sistema que funciona con PipeWire.

<sup>1</sup>software para audio de escritorio/bloatware

<sup>2</sup>audio profesional de baja latencia

<sup>3</sup>Asegúrate de que pulseaudio-alsa, el paquete que redirige ALSA a PulseAudio, está desinstalado.

<sup>4</sup>Advanced Linux Sound Architecture[22].



### 5.2.2. Integración con herramientas de PulseAudio

Tras la ejecución anterior, podemos empezar a utilizar los comandos que usaríamos en un sistema con PulseAudio, si se ha instalado correctamente el siguiente comando debería de funcionar perfectamente:

```
1 $ pactl info
```

En específico, en **ServerName** debería de aparecer el valor **PulseAudio (on PipeWire X.Y.Z)**. A su vez, ya podemos ejecutar otros comandos de pactl, tales como el control de volumen:

```
1 $ pactl set-sink-volume @DEFAULT_SINK@ +3% #subir
2 $ pactl set-sink-volume @DEFAULT_SINK@ -3% #bajar
3 $ pactl set-sink-mute @DEFAULT_SINK@ toggle #mute
```

Los cuales ya podemos utilizar para la configuración de i3wm:

```
1 bindsym XF86AudioRaiseVolume exec --no-startup-id pactl set-sink-volume
  ↳ @DEFAULT_SINK@ +3% && $refresh_i3status
2 bindsym XF86AudioLowerVolume exec --no-startup-id pactl set-sink-volume
  ↳ @DEFAULT_SINK@ -3% && $refresh_i3status
3 bindsym XF86AudioMute exec --no-startup-id pactl set-sink-mute @DEFAULT_SINK@
  ↳ toggle && $refresh_i3status
```

### 5.2.3. Bluetooth con PipeWire y aplicaciones recomendadas

Esta sección existe no solo para aprovechar e incluir la instalación de bluetooth y su configuración<sup>5</sup> con PipeWire, sino que se usará para recomendar dos GUIs muy útiles a la hora de gestionar las conexiones bluetooth y el audio.

Por un lado, para instalar bluetooth hay que asegurarse de que están instalados los paquetes bluez y bluez-utils. Tras ello, hay que habilitar el servicio a nivel de **sistema** de bluetooth:

```
1 $ systemctl enable --now bluetooth
```

Tras ello, podemos utilizar **bluetoothctl** para establecer conexiones por cli:

```
1 $ bluetoothctl
2 $ [bluetoothctl] power on
3 $ [bluetoothctl] scan on #aparecerán dispositivos con su MAC
4 $ [bluetoothctl] pair XX:XX:XX:XX:XX:XX
5 $ [bluetoothctl] connect XX:XX:XX:XX:XX:XX
6 $ [bluetoothctl] trust XX:XX:XX:XX:XX:XX
```

En cuanto a las aplicaciones que se recomiendan, por un lado, está bluetui, una TUI para gestionar las conexiones bluetooth, por otro lado, para el gestión de audio de entrada y salida se recomienda pavucontrol<sup>6</sup>.

<sup>5</sup>Que en la mayoría de casos es automática.

<sup>6</sup>No se puede hacer el suficiente énfasis en lo útil que es esta herramienta

## 6. Ricing y estética

### 6.1. .Xresources y notify\_send

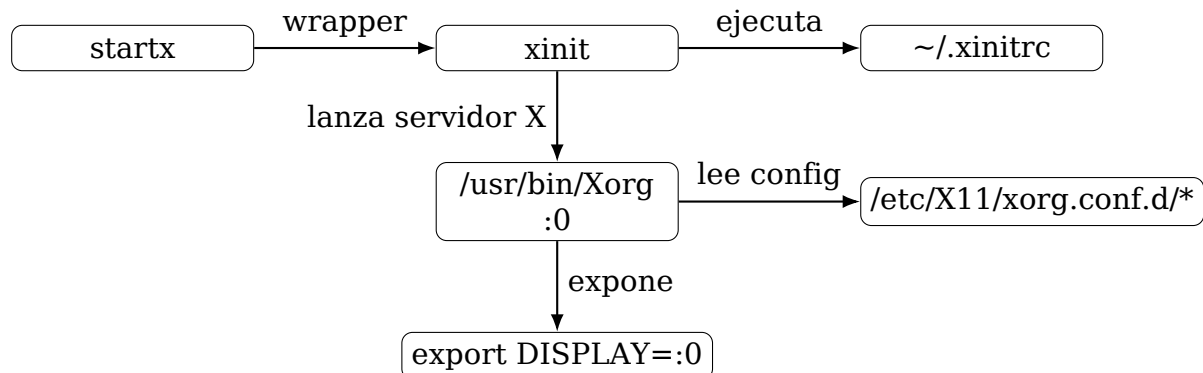
#### 6.1.1. Instalación de notify-send y dunst

Se recuerda al lector que es libre de elegir el servidor de notificaciones que prefiera, pero, en este caso, se ha optado por utilizar dunst por familiaridad y porque funciona bastante bien por defecto.

Por un lado, hay que instalar los paquetes `libnotify` y `dunst`, el primero sirve para tener **notify-send** disponible en el sistema, el segundo instala el servidor de notificaciones. Tras ello, se puede configurar el servidor mediante la configuración del archivo `/.config/dunst/dunstrc`. Como ya se ha indicado, la configuración por defecto es bastante buena para probar el servicio, esta configuración se encuentra en `/usr/share/dunst/dunstrc`. En la siguiente parte de la sección se muestra una forma de configurar donde se lanza dunst.

#### 6.1.2. startx y como se lanzan los servidores X11

Antes de empezar a configurar .Xresources, es necesario destacar y entender como funciona **startx**. Este es el diagrama de ejecución de startx:



Dicho en verso, cuando lanzamos `startx` lo que en verdad se está haciendo es lanzar un `wrapper` de `xinit`<sup>[23]</sup>, este `wrapper` busca el script de inicialización del usuario, normalmente llamada `~/.xinitrc` y, tras ello, lanza el servidor X (ejecutando Xorg) junto a la inicialización que haya encontrado.

Una vez se lanza el servidor X, este lee los archivos de configuración creados sobre dispositivos que se encuentran en `/etc/X11/xorg.conf`<sup>1</sup>, tras lo cual, va ejecutando la configuración lexicográficamente y, a su vez, detecta el hardware conectado en el sistema<sup>2</sup>.

En este caso, se destaca .Xresources porque son recursos X que no se añaden automáticamente con **xinit**, a continuación se incluyen un ejemplo de configuración con explicaciones:

<sup>1</sup>Se ha obviado porque no es muy recomendable poner la configuración ahí

<sup>2</sup>Monitores, teclados, etc. Esto solo lo hace una vez al lanzar el servidor

```

1 Xft.dpi:192
2 Xcursor.size: 32
3 Xcursor.theme: Bibata-Modern-Classic

```

- **Xft.dpi**: Permite definir el dpi<sup>3</sup> que usarán las aplicaciones gráficas.
- **Xcursor.size**: Define el tamaño del cursor.
- **Xcursor.theme**: Elige el tema utilizado por el cursor.

Cabe destacar que el dpi definido es adecuado para pantallas con resoluciones 2K, 3K y quizás 4K. Para pantallas full HD se recomienda usar 90 o 92 de dpi.

Este archivo se cargará usando el comando **xrdb** en `.xinitrc`, siendo un ejemplo de configuración el que sigue:

```

1 xrandr --dpi 192 &
2 xrdb -merge ~/.Xresources &
3 picom &
4 xset r 250 30 &
5 dunst &
6 exec i3;

```

Quizás el lector se esté preguntando porqué estamos usando la opción `-dpi 192` si ya hemos definido el dpi en `/.Xresources`. Esto se debe a que muchas aplicaciones antiguas dependen de que la definición de la dpi se encuentre en `xrandr`. Lo único que se ha hecho con estas configuraciones es definir el dpi en dos posibles niveles.

Cabe recordar que `.xinitrc` es un script, por lo que estamos haciendo es ejecutar comandos, primero definimos el dpi a nivel de **xrandr**, después incluimos la configuración hecha en `/.Xresources`, tras ello, lanzamos el compositor **picom**. Con **xset** establecemos que a los 250 milisegundos de mantener una tecla presionada se empiece a repetir la tecla 30 veces por segundo. Lanzamos el servidor de notificaciones **dunst** y por último usamos **exec** para lanzar `i3wm`.

Tras ello, podemos probar si el servidor de notificaciones se ha lanzado correctamente con el siguiente comando:

```

1 $ notify-send "prueba"

```

## 6.2. Wallpaper, bar (i3bar) y launchers (rofi)

Vamos a empezar con el wallpaper, por un lado tenemos `nitrogen`, que está dedicado para establecer fondos de pantalla. Sin embargo, existe un programa más flexible que, además de permitir cambiar fondos de pantalla, se puede utilizar para otras partes de la configuración de `i3wm`<sup>4</sup>.

<sup>3</sup>dots per inch

<sup>4</sup>Visualización de imágenes, bastante más útil de lo que parece.

En este caso, tras haber instalado el paquete, bastaría con añadir la siguiente línea a `/.config/i3/config`:

```
1 exec --no-startup-id feh --bg-scale
  ↳ ~/.local/share/wallpapers/imagen_que_va_a_ser_wallpaper.jpg
```

En el proceso se creará el archivo `/.fehbg`.

Tras ello, podemos pasar a configurar la barra de estado, por defecto i3wm usa **i3bar** con una configuración básica<sup>5</sup> Podemos cambiar esta configuración para usar i3status o, directamente, para usar polybar. A continuación un par de ejemplos de dicha configuración:

```
1 #si se piensa utilizar i3bar, dentro de bar {
2 status_command i3blocks -c ~/.i3/i3blocks.conf #ejemplo usando i3blocks.conf en un
  ↳ directorio específico
3 status_command i3status -c ~/.i3/i3status.conf #ejemplo usando usando i3status,
  ↳ mucho más recomendable que blocks
4
5 #en cambio si se está usando polybar, se puede usar usar un script como wrapper o
  ↳ lanzarlo directamente
6 #pero es preferible el wrapper
7 bindsym $mod+b exec bash -c '~/.config/polybar/launch.sh'
```

A continuación se incluye un ejemplo de configuración de i3status.conf:

```
1 # i3status configuration file.
2 # see "man i3status" for documentation.
3
4 # It is important that this file is edited as UTF-8.
5 # The following line should contain a sharp s:
6 # s
7 # If the above line is not correctly displayed, fix your editor first!
8
9 general {
10     colors = false
11     interval = 5
12 }
13
14 order += "wireless _first_"
15 order += "ethernet _first_"
16 order += "battery all"
17 #order += "load"
18 #order += "memory"
19 order += "volume master"
20 order += "tztime local"
21
22
23 wireless _first_ {
24     format_up = "    %quality at %essid (%ip) "
25     format_down = "    "
26 }
27
```

<sup>5</sup>Diseñada especialmente para incluir cosas que podrían ser útiles.

```
28 ethernet _first_ {
29     format_up = "    %ip (%speed) "
30     format_down = ""
31 }
32
33 battery all {
34     format = "%status %percentage %remaining "
35     status_bat = " "
36     status_chr = " "
37 }
38
39 disk "/" {
40     format = "%avail"
41 }
42
43 load {
44     format = "%1min"
45 }
46
47 memory {
48     format = "%used | %available"
49     threshold_degraded = "1G"
50     format_degraded = "MEMORY < %available"
51 }
52
53 tztime local {
54     format = "    %a, %b %d - %H:%M "
55 }
56
57 volume master {
58     format = "Vol %volume"
59     format_muted = "Muted %volume"
60     device = "default"
61 }
```

A pesar de que la filosofía de este documento es tener toda la documentación en un único pdf, las configuraciones de polybar pueden ser especialmente largas<sup>6</sup>. Por lo que se ha optado no incluir una configuración de polybar al manual.

---

<sup>6</sup>Y como nota adicional, el autor es pésimo configurando polybar, por lo que no recomienda usar sus ejemplos.

## 7. Solución de problemas y checklist

### 7.1. Logs relevantes (journalctl)

Cuando algo no funciona, lo primero que se debería de hacer es mirar los logs del sistema para comprobar que ha pasado. En Arch Linux, debido al uso de systemd, la mayoría de los registros pasan por journald[24], los logs específicos de Xorg son adicionales.

#### 7.1.1. journalctl

**journalctl** es la herramienta utilizada para consultar el journal de systemd. También es posible usar **systemctl status servicio** para echar un ojo rápido a los logs de un servicio en concreto.

A continuación se añaden ejemplos de ejecución de journalctl:

```
1 $ journalctl -b #logs del arranque actual
2 $ journalctl -b 1 #logs del arranque anterior
3 $ journalctl -f #seguir los logs en tiempo real
4 # filtros por servicio
5 $ journalctl -u NetworkManager
6 $ journalctl -u pipewire --user
7 $ journalctl -p err -b #solo errores
8 $ journalctl -p warning -b #solo advertencias
9 $ journalctl --since "1 hour ago" #logs de hace 1 hora
```

Añadir **-e** para ir directamente al final del log y **-n 50** para ver solo las últimas 50 líneas. La combinación **journalctl -xe** es muy común para diagnóstico rápido tras un error.

#### 7.1.2. Xorg.0.log (solo X11)

Cuando i3wm no arranca o la pantalla se queda en negro, el log de Xorg es el primer lugar donde mirar:

```
1 # Log de la sesión X actual
2 $ cat /var/log/Xorg.0.log
3 # Filtrar solo errores y advertencias
4 $ grep -E "\(EE\)|\(WW\)" /var/log/Xorg.0.log
```

Las líneas marcadas con **(EE)** son errores; las marcadas con **(WW)** son advertencias. La mayoría de las advertencias son inofensivas, pero los errores suelen indicar problemas con drivers o dispositivos.

Si se usa **startx** en lugar de un display manager, el log también puede encontrarse en `/.local/share/xorg/Xorg.0.log`.

## 7.2. Problemas frecuentes

### 7.2.1. Como puedo establecer el teclado de mi idioma a español

Para ello es necesario añadir un archivo en `/etc/X11/xorg.conf.d/`, por ejemplo `00-keyboard.conf`, con la siguiente configuración:

```
1 Section "InputClass"
2     Identifier "system-keyboard"
3     MatchIsKeyboard "on"
4     Option "XkbLayout" "es"
5 EndSection
```

De esta manera todo teclado que se conecte se considerará un teclado en español. Es posible que haya que hacer pequeños retoques en la configuración, como elegir el layout `pc-105` entre otros. Sobre todo si se usan teclados con layouts extraños.

### 7.2.2. Como habilitar el tap para hacer click en el touchpad

Para ello es necesario añadir un archivo en `/etc/X11/xorg.conf.d/`, por ejemplo `70-synaptics.conf`, con la siguiente configuración:

```
1 Section "InputClass"
2     Identifier "touchpad"
3     Driver "synaptics"
4     MatchIsTouchpad "on"
5     Option "TapButton1" "1"
6     Option "TapButton2" "2"
7     Option "TapButton3" "3"
8     Option "tapping" "on"
9 EndSection
```

A su vez, se ha añadido la configuración de los botones<sup>1</sup>. Una vez configurado y tras un reinicio, el touchpad debería de permitir hacer taps.

### 7.2.3. Como hacer hot-plugging de múltiples monitores en X11

Esto es mucho más complicado que los problemas anteriores, principalmente porque X11 no fue diseñado con soporte multi monitor en cuenta, la opción más simple consiste en el siguiente esquema:

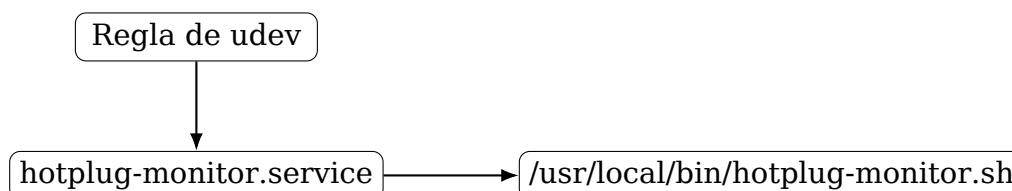


Figura 7.1.: Flujo desde la regla udev hasta el hotplug de monitores

Empezamos definiendo una regla udev en `/etc/udev/rules.d/85-drm-hotplug.rules`, que contendrá lo siguiente:

<sup>1</sup>El por defecto está bien configurado, pero es mejor prevenir que curar.

```

1 ACTION=="change", SUBSYSTEM=="drm", ENV{HOTPLUG}=="1", TAG+="systemd",
  ↳ ENV{SYSTEMD_WANTS}+="hotplug-monitor.service"

```

Básicamente estamos diciéndole a udev que avise a systemd de que se ha conectado un monitor y que le interesa lanzar el servicio **hotplug-monitor.service**.

Tras ello, definimos el servicio hotplug-monitor en /etc/systemd/system/hotplug-monitor.service, con el siguiente contenido:

```

1 [Unit]
2 Description=Reconfigure X monitors
3 ConditionPathExists=/tmp/.X11-unix/X0
4
5 [Service]
6 Type=oneshot
7 Environment=DISPLAY=:0
8 Environment=DBUS_SESSION_BUS_ADDRESS=unix:path=/run/user/1000/bus
9 Environment=XAUTHORITY=/home/TU_NOMBRE_DE_USUARIO/.Xauthority
10 ExecStart=/usr/local/bin/hotplug-monitor.sh

```

Se recuerda al lector de que se asegure de que las variables de entorno tienen los valores correctos una vez haya copiado el .service.

Y, como punto final, se define el script /usr/local/bin/hotplug-monitor.sh:

```

1 #!/usr/bin/env bash
2 # Sends X display to an external monitor, with automatic scale/fb/panning
3 ↳ calculation
4 # Based on user's original script, extended to compute --scale, --fb and --panning.
5 set -euo pipefail
6
7 STATEDIR="/run/monitor-hotplug"
8 STATEFILE="$STATEDIR/state"
9 LOCKFILE="$STATEDIR/lock"
10 INTERNAL="eDP-1" #cambiar si es necesario
11
12 WAIT_TRIES=20
13 WAIT_DELAY=0.1
14
15 mkdir -p "$STATEDIR"
16 exec 9>"$LOCKFILE"
17 if ! flock -n 9; then
18     echo "Another instance is running; exiting."
19     exit 0
20 fi
21
22 if [ -n "${WAYLAND_DISPLAY:-}" ]; then
23     echo "Wayland detected. This script is for X11. Exiting"
24     exit 0
25 fi
26
27 wait_for_outputs() {
28     local tries="$WAIT_TRIES"
29     local delay="$WAIT_DELAY"

```



```

29
30 while (( tries-- > 0 )); do
31     local ok=1
32     for out in "${CONNECTED[@]}"; do
33         xrandr | grep -qE "^$out connected" || ok=0
34     done
35     (( ok )) && return 0
36     sleep "$delay"
37 done
38 return 1
39 }
40
41 CURRENTSTATE="$(for s in /sys/class/drm/card*/status; do
42     conn="$(basename "$(dirname "$s")" )" #get the connector name
43     case "$conn" in
44         *eDP*) continue ;; #ignore internal
45     esac
46     printf "%s=%s\n" "$conn" "$(cat "$s")"
47 done | sort || true)"
48
49 if [ -f "$STATEFILE" ]; then
50     OLDSTATE="$(cat "$STATEFILE")"
51 else
52     OLDSTATE=""
53 fi
54
55 printf "%s\n" "$CURRENTSTATE" > "$STATEFILE"
56
57 [ "$CURRENTSTATE" = "$OLDSTATE" ] && exit 0
58
59 echo "Changes detected, applying..."
60
61 CONNECTED=()
62 XRANDR_CMD=(xrandr --output "$INTERNAL" --off)
63
64 while IFS='=' read -r conn status; do
65     conn_cardless="${conn#card*-}"
66
67     case "$conn_cardless" in
68         HDMI-A-*) conn_cardless="HDMI-${conn_cardless#HDMI-A-}" ;;
69         #add other mapping in case drivers use a different fragment
70     esac
71     if [ "$status" == "connected" ]; then
72         CONNECTED+=("$conn_cardless")
73     else
74         XRANDR_CMD+=(--output "$conn_cardless" --off)
75     fi
76 done <<< "$CURRENTSTATE"
77
78 wait_for_outputs || { echo "Timed out waiting for xrandr"; exit 0; }
79 "${XRANDR_CMD[@]}"
80 # Helper: get current mode (width height) for an output
81 get_mode() {
82     local out="$1"
83     xrandr | awk -v o="$out" '
84         new_output {new_output=0
85             for(i=1;i<=NF;i++){
86                 if (match($i, /^[0-9]+x[0-9]+(\+[0-9]+){2}$/)) {

```

```

87         split($i,a,"+")
88         split(a[1],wh,"x")
89         print wh[1],wh[2]
90         exit
91     } else if (match($i, /^[0-9]+x[0-9]+$/)) {
92         split($i,wh,"x")
93         print wh[1],wh[2]
94         exit
95     }
96 }
97 }
98 $1==o && $2=="connected" {new_output=1}'
99 }
100
101 # Get internal native resolution
102 INTERNAL_WH="$(get_mode "$INTERNAL")"
103 if [ -z "$INTERNAL_WH" ]; then
104     echo "Failed to detect internal mode for $INTERNAL; falling back to --auto for
↪ internal"
105     xrandr --output "$INTERNAL" --auto --primary
106     exit 0
107 fi
108
109 INTERNAL_W=$(echo "$INTERNAL_WH" | awk '{print $1}')
110 INTERNAL_H=$(echo "$INTERNAL_WH" | awk '{print $2}')
111 echo "Internal resolution: ${INTERNAL_W}x${INTERNAL_H}"
112
113 # Turn off detected externals (clean start)
114 for out in "${CONNECTED[@]}"; do
115     echo "Disabling $out"
116     xrandr --output "$out" --off || true
117 done
118
119 # Build arrays for external data
120 declare -a EXT_W
121 declare -a EXT_H
122 declare -a EXT_SCALE
123 declare -a EXT_SCALED_W
124 declare -a EXT_SCALED_H
125 declare -a EXT_POS_X
126
127 # Collect information and compute scales
128 total_left_width=0
129 max_height="$INTERNAL_H"
130
131 for idx in "${!CONNECTED[@]}"; do
132     out="${CONNECTED[$idx]}"
133     mode="$(get_mode "$out")"
134     EXT_POS_X[$idx]=$total_left_width
135     if [ -z "$mode" ]; then
136         echo "Warning: cannot find current mode for $out, using --auto later"
137         EXT_W[$idx]=0
138         EXT_H[$idx]=0
139         EXT_SCALE[$idx]=1
140         EXT_SCALED_W[$idx]=0
141         EXT_SCALED_H[$idx]=0
142         continue
143     fi

```

```

144 w=$(echo "$mode" | awk '{print $1}')
145 h=$(echo "$mode" | awk '{print $2}')
146 # compute scale = internal_width / ext_width (so logical widths match)
147 scale=$(awk -v iw="$INTERNAL_W" -v ew="$w" 'BEGIN{if(ew==0){print 1; exit}
↪ s=iw/ew; if(s<1) s=1; printf("%.8f", s)}')
148 # clamp sensible values (avoid ridiculous transforms)
149 scale=$(awk -v s="$scale" 'BEGIN{ if(s<1) s=1; if(s>4) s=4; printf("%.8f", s)}')
150 # compute scaled pixel sizes and round to integer
151 scaled_w=$(awk -v w="$w" -v s="$scale" 'BEGIN{printf("%d", (w*s)+0.5)}')
152 scaled_h=$(awk -v h="$h" -v s="$scale" 'BEGIN{printf("%d", (h*s)+0.5)}')
153 EXT_W[$idx]=$w
154 EXT_H[$idx]=$h
155 EXT_SCALE[$idx]=$scale
156 EXT_SCALED_W[$idx]=$scaled_w
157 EXT_SCALED_H[$idx]=$scaled_h
158 total_left_width=$(( total_left_width + scaled_w ))
159 if (( scaled_h > max_height )); then max_height=$scaled_h; fi
160 echo "Detected $out native ${w}x${h} -> scale ${scale} -> logical
↪ ${scaled_w}x${scaled_h}"
161 done
162
163 # Internal position is the sum of all externals' scaled widths (we place externals
↪ to the left)
164 INTERNAL_POS_X=$total_left_width
165 # Total framebuffer dimensions: internal pos + internal width, and max height
166 FB_W=$(( INTERNAL_POS_X + INTERNAL_W ))
167 FB_H=$max_height
168
169 # add 1px guard to work around pointer boundary rounding bugs in some drivers
170 FB_W=$(( FB_W + 1 ))
171 FB_H=$(( FB_H + 1 ))
172
173 # Build xrandr command: set framebuffer, place internal, then externals in order
174 XRANDR_CMD=(xrandr --fb "${FB_W}x${FB_H}")
175
176 # Configure internal explicitly (keep as primary)
177 XRANDR_CMD+=(--output "$INTERNAL" --mode "${INTERNAL_W}x${INTERNAL_H}" --pos
↪ "${INTERNAL_POS_X}x0" --primary)
178 LAST_ONE="$INTERNAL"
179 # Configure each external
180 for idx in "${!CONNECTED[@]"; do
181     out="${CONNECTED[$idx]}"
182     w="${EXT_W[$idx]}"
183     h="${EXT_H[$idx]}"
184     scale="${EXT_SCALE[$idx]}"
185     scaled_w="${EXT_SCALED_W[$idx]}"
186     scaled_h="${EXT_SCALED_H[$idx]}"
187     posx="${EXT_POS_X[$idx]}"
188
189     if [ "$w" -eq 0 ]; then
190         # unknown mode -> try auto, place to computed pos but without scale/panning
191         XRANDR_CMD+=(--output "$out" --auto --pos "${posx}x0")
192     else
193         # add the output with scale and panning to reserve the logical area
194         XRANDR_CMD+=(--output "$out" --mode "${w}x${h}" --scale "${scale}x${scale}"
↪ --pos "${posx}x0" --panning "${scaled_w}x${scaled_h}+${posx}+0")
195     fi
196     LAST_ONE=$out

```

```

197 done
198
199 echo "Running: ${XRANDR_CMD[*]}"
200
201 "${XRANDR_CMD[@]}"
202
203 if [ "${#CONNECTED[@]}" -gt 0 ]; then #driver bug workaround for HW cursors
204     xrandr --output "$INTERNAL" --scale 0.9999x0.9999
205 fi
206
207 # final xrdp / X resources hint: some toolkits require Xft.dpi update to reflow
208 ↪ fonts for old apps
209 # xrdp -merge ~/.Xresources # uncomment if you manage DPI with Xresources, ignorao
210 echo "Done."

```

El script es bastante largo, es de realización propia y está en inglés. Hay cosas a mejorar como por ejemplo que hay un error en el cálculo del framebuffer, calculando siempre las dimensiones correctas -1. También habría que buscar opciones para integrarlo de mejor manera para permitir al usuario colocar los monitores donde él quiera, porque por ahora el script solo los va poniendo todos a la izquierda.

Tras ello, se pueden actualizar los servicios y reglas del sistema:

```

1 $ systemctl daemon-reload
2 $ udevadm control --reload-rules
3 $ udevadm trigger

```

Ya solo quedaría comprobar que funciona y mirar los logs en caso de que no sea así. Puede tardar un poco en actualizar todos los monitores.

## A. Configuración i3wm en portátil

```
1  # xprop to check window
2  # xmodmap [ --pke ] view keymap
3  #####
4
5
6  #####
7  ##### IMPORTANT SETTINGS #####
8  #####
9
10 ##### Alt and super key, in their respective order
11 set $mod Mod1
12 set $alt_mod Mod4
13
14 ##### reload the configuration file
15 bindsym $mod+Shift+c reload
16 ##### restart i3 inplace (preserves your layout/session, can
17   ↳ be used to upgrade i3)
18 bindsym $mod+Shift+r restart
19
20 ##### title fonts
21 font pango:Iosevka 11
22
23 #####gaps
24 #for_window [class="^.*"] border normal [DEPRECATED]
25 #for_window [class="^.*"] border pixel 2
26 default_border pixel 2
27 default_floating_border pixel 2
28 set $definner 4
29 set $defouter 2
30 set $floatingx 8
31 set $floatingy 28
32 gaps inner $definner
33 gaps outer $defouter
34
35 #####
36 ##### WORKSPACE SETTINGS #####
37 #####
38 #####
39
40 #####
41 ##### WORKSPACE NAMING #####
42 #####
43
44 #OLD SETUP
45 #set $tag1 "1"
46 #set $tag2 "2"
47 #set $tag3 "3"
48 #set $tag4 "4"
49 #set $tag5 "5"
```

```

50 #set $tag6 "6 "
51 #set $tag7 "7 "
52 #set $tag8 "8 "
53 #set $tag9 "9 "
54 #set $tag0 "0 "
55 #set $tag11 "11 F1"
56 #set $tag12 "12 F2"
57 #set $tag13 "13 F3"
58 #set $tag14 "14 F4"
59 #set $tag15 "15 F5"
60 #set $tag16 "16 F6"
61 #set $tag17 "17 F7"
62 #set $tag18 "18 F8"
63 #set $tag19 "19 F9"
64 #set $tag20 "20 F10"
65
66 #changed to work with polybar
67 set $tag1 "1"
68 set $tag2 "2"
69 set $tag3 "3"
70 set $tag4 "4"
71 set $tag5 "5"
72 set $tag6 "6"
73 set $tag7 "7"
74 set $tag8 "8"
75 set $tag9 "9"
76 set $tag0 "0"
77 set $tag11 "11"
78 set $tag12 "12"
79 set $tag13 "13"
80 set $tag14 "14"
81 set $tag15 "15"
82 set $tag16 "16"
83 set $tag17 "17"
84 set $tag18 "18"
85 set $tag19 "19"
86 set $tag20 "20"
87
88 #####
89
90 bindsym $mod+1 workspace $tag1
91 bindsym $mod+2 workspace $tag2
92 bindsym $mod+3 workspace $tag3
93 bindsym $mod+4 workspace $tag4
94 bindsym $mod+5 workspace $tag5
95 bindsym $mod+6 workspace $tag6
96 bindsym $mod+7 workspace $tag7
97 bindsym $mod+8 workspace $tag8
98 bindsym $mod+9 workspace $tag9
99 bindsym $mod+0 workspace $tag0
100
101 ##### More workspaces, not used much but useful when needed
102 #bindsym $mod+F1 workspace $tag11
103 #bindsym $mod+F2 workspace $tag12
104 #bindsym $mod+F3 workspace $tag13
105 #bindsym $mod+F4 workspace $tag14
106 #bindsym $mod+F5 workspace $tag15
107 #bindsym $mod+F6 workspace $tag16

```

```

108 #bindsym $mod+F7 workspace $tag17
109 #bindsym $mod+F8 workspace $tag18
110 #bindsym $mod+F9 workspace $tag19
111 #bindsym $mod+F10 workspace $tag20
112
113 #####
114 ##### WORKSPACE RULES #####
115 #####
116
117 ##### WORKSPACE MOVEMENT #####
118
119 #####move container to workspace
120 bindsym $mod+Shift+1 move container to workspace $tag1
121 bindsym $mod+Shift+2 move container to workspace $tag2
122 bindsym $mod+Shift+3 move container to workspace $tag3
123 bindsym $mod+Shift+4 move container to workspace $tag4
124 bindsym $mod+Shift+5 move container to workspace $tag5
125 bindsym $mod+Shift+6 move container to workspace $tag6
126 bindsym $mod+Shift+7 move container to workspace $tag7
127 bindsym $mod+Shift+8 move container to workspace $tag8
128 bindsym $mod+Shift+9 move container to workspace $tag9
129 bindsym $mod+Shift+0 move container to workspace $tag0
130
131 #####move container to second layer
132 #bindsym $mod+Shift+F1 move container to workspace $tag11
133 #bindsym $mod+Shift+F2 move container to workspace $tag12
134 #bindsym $mod+Shift+F3 move container to workspace $tag13
135 #bindsym $mod+Shift+F4 move container to workspace $tag14
136 #bindsym $mod+Shift+F5 move container to workspace $tag15
137 #bindsym $mod+Shift+F6 move container to workspace $tag16
138 #bindsym $mod+Shift+F7 move container to workspace $tag17
139 #bindsym $mod+Shift+F8 move container to workspace $tag18
140 #bindsym $mod+Shift+F9 move container to workspace $tag19
141 #bindsym $mod+Shift+F10 move container to workspace $tag20
142
143 ##### [move to] new workspace NOT USED
144 #bindsym $mod+i exec i3-input -F "workspace \"%s\"" -P 'Workspace: '
145 #bindsym $mod+Shift+i exec i3-input -F "move container to workspace \"%s\"" -P
    ↪ 'Move to workspace: '
146
147 #####move IN right left NOT USED
148 #bindsym $mod+Control+l workspace next
149 #bindsym $mod+Control+h workspace prev
150
151 #####move TO right left NOT USED
152 #bindsym $mod+Shift+Control+Right move to workspace next
153 #bindsym $mod+Shift+Control+Left move to workspace prev
154
155 ##### WORKSPACE POSITION #####
156
157 #####Configure where are they going to display
158
159 #odd workspaces, these won't change
160 workspace $tag1 output eDP-1
161 workspace $tag3 output eDP-1
162 workspace $tag5 output eDP-1
163 workspace $tag7 output eDP-1
164 workspace $tag9 output eDP-1

```

```

165 workspace $tag11 output eDP-1
166 workspace $tag13 output eDP-1
167 workspace $tag15 output eDP-1
168 workspace $tag17 output eDP-1
169 workspace $tag19 output eDP-1
170
171 #dual monitor setup with even workspaces
172 workspace $tag2 output HDMI-1
173 workspace $tag4 output HDMI-1
174 workspace $tag6 output HDMI-1
175 workspace $tag8 output HDMI-1
176 workspace $tag0 output HDMI-1
177 workspace $tag12 output HDMI-1
178 workspace $tag14 output HDMI-1
179 workspace $tag16 output HDMI-1
180 workspace $tag18 output HDMI-1
181 workspace $tag20 output HDMI-1
182
183
184 ##### WORKSPACE ASSIGNATION #####
185
186 assign [class="Alacritty"] $tag1
187 assign [class="firefox"] $tag3
188 #assign [class="Chromium"] $tag4
189 assign [class="elecwhat"] $tag4
190 #assign [class="Thunar"] $tag5
191 #assign [class="Atril"] $tag6
192 assign [class="TelegramDesktop"] $tag8
193 assign [class="discord"] $tag8
194 #assign [class="Solaar"] $tag0
195 #assign [class="Spotify"] $tag0
196 assign [class="pavucontrol"] $tag0
197 assign [class="steam"] $tag9
198
199 #####HTOP FOR THE WIN
200 for_window [class="Alacritty" title="~htop$"] floating enable
201 for_window [class="Alacritty" title="~htop$"] resize set 1200 720
202 for_window [class="Alacritty" title="~htop$"] move position $floatingx $floatingy
203 for_window [class="Alacritty" title="~htop$"] sticky enable
204
205 #####NVTOP FOR THE WIN
206 for_window [class="Alacritty" title="~nvtop$"] floating enable
207 for_window [class="Alacritty" title="~nvtop$"] resize set 1200 720
208 for_window [class="Alacritty" title="~nvtop$"] move position $floatingx 748
209 for_window [class="Alacritty" title="~nvtop$"] sticky enable
210
211 #####feh for screenshots
212 for_window [class="quick-snap"] floating enable
213 for_window [class="quick-snap"] move position center
214 for_window [class="quick-snap"] sticky enable
215 for_window [class="quick-snap"] border pixel 2
216
217 #####fzf for clipboard with alacritty
218 #for_window [class="Alacritty" title="~clipboard$"] floating enable
219 #for_window [class="Alacritty" title="~clipboard$"] resize set 800 360
220 #for_window [class="Alacritty" title="~clipboard$"] move position center
221 #for_window [class="Alacritty" title="~clipboard$"] sticky enable
222 #for_window [class="Alacritty" title="~clipboard$"] focus

```



```

223
224 #####gaming chair
225 for_window [class="steam"] floating enable
226 for_window [class="steam" title="^Steam$"] floating disable
227
228 #####bluetooth
229 #for_window [class="Blueberry.py"] floating enable
230 #for_window [class="Blueberry.py"] sticky enable
231
232 #####spotify+spicify error with https, ignore and move the cava
233 ↪ window to the correct position
234 for_window [class="zenity" title="^Error$"] kill, exec --no-startup-id sleep 0.5 &&
235 ↪ ~/.i3/bin/cava-reajust.sh $tag0
236
237 #####
238 ##### CONTAINER SETTINGS #####
239 #####
240
241 #####
242 ##### COLOR SETTINGS #####
243 #####
244
245
246 # define colors
247 #abbaddon color
248 set $abbaddon_col #231f20
249 #macragge blue
250 set $macragge_blue #0d407f
251 #zandri dust
252 set $zandri_dust #9e915c
253 #light french blue
254 set $l_french_blue #4380b6
255 #retributor armour
256 set $retrib_armour #c39e81
257 #ulthuan grey
258 set $ulthuan_grey #c7e0d9
259 #dombull brown
260 set $dombull_brown #5d0009
261 #celestra grey
262 set $celestra_grey #90a8a8
263 #ushabti bone
264 set $ushabti_bone #bbbb7f
265
266 set $nord_background #2e3440
267 set $nord_foreground #D8DEE9
268
269 set $nord1 #3B4252
270 set $nord2 #434C5E
271 set $nord3 #4C566A
272 set $nord5 #E5E9F0
273 set $nord8 #88C0D0
274 set $nord10 #5E81AC
275 set $nord15 #B48EAD
276
277 # set some nice colors border background text indicator
278 client.focused $macragge_blue $l_french_blue $abbaddon_col $celestra_grey

```

```

279 client.focused_inactive $zandri_dust $ushabti_bone $zandri_dust $celestra_grey
280 client.unfocused        $celestra_grey $macragge_blue $ushabti_bone $celestra_grey
281 client.urgent           $celestra_grey $dombull_brown $retrib_armour $celestra_grey
282
283
284 #####
285 ##### SYSTEM SETTINGS #####
286 #####
287
288 ##### BASIC COMMANDS #####
289
290 ##### kill focused window
291 bindsym $mod+Shift+q kill
292
293 ##### start rofi (a program launcher), default is dmenu
294 #bindsym $mod+d exec dmenu_run
295 bindsym $mod+d exec --no-startup-id rofi -show run -theme
296 ↪ ~/.config/rofi/config.rasi
297 bindsym $alt_mod+d exec --no-startup-id rofi -show drun
298 bindsym $mod+Shift+d exec --no-startup-id rofi -show emoji
299 bindsym $mod+c exec --no-startup-id rofi -show calc -no-show-match
300 ↪ -no-persist-history
301
302 ##### Use Mouse+$mod to drag floating windows to their wanted
303 ↪ position
304 floating_modifier $mod
305
306 ##### Workspace toggle $mod + 'q'
307 bindsym $mod+q workspace back_and_forth
308
309
310 ##### Border toggle
311 bindsym $mod+x border toggle
312
313
314 ##### Screenshot
315 ### Permanent screenshots
316 bindsym Print exec bash -c 'exec maim -u -f png "$(xdg-user-dir
317 ↪ PICTURES)/screenshots/$(date +%F_%T).png"'
318 bindsym Shift+Print exec bash -c 'exec maim -u -f png -s "$(xdg-user-dir
319 ↪ PICTURES)/screenshots/$(date +%F_%T).png"'
320 ### Clipboard screenshots
321 bindsym $alt_mod+Print exec maim -u -f png | xclip -selection clipboard -target
322 ↪ image/png
323 bindsym $alt_mod+Shift+Print exec maim -u -f png -s | xclip -selection clipboard
324 ↪ -target image/png
325 ### Quick-snaps
326 bindsym Ctrl+Print exec maim -u -f png -m 10 -s /tmp/quick-snap.png && feh --class
327 ↪ quick-snap /tmp/quick-snap.png
328
329 ##### CONTROL COMMANDS #####
330
331 ##### Pulse Audio controls
332 set $refresh_i3status killall -SIGUSR1 i3status
333 bindsym XF86AudioRaiseVolume exec --no-startup-id pactl set-sink-volume
334 ↪ @DEFAULT_SINK@ +3% && $refresh_i3status
335 bindsym XF86AudioLowerVolume exec --no-startup-id pactl set-sink-volume
336 ↪ @DEFAULT_SINK@ -3% && $refresh_i3status
337 bindsym XF86AudioMute exec --no-startup-id pactl set-sink-mute @DEFAULT_SINK@
338 ↪ toggle && $refresh_i3status

```

```

327 ##### Media player controls
328 bindsym XF86AudioPlay exec mpc toggle
329 bindsym XF86AudioPause exec mpc pause
330 bindsym XF86AudioNext exec mpc next
331 bindsym XF86AudioPrev exec mpc prev
332
333
334 #bindsym XF86Launch1 exec xset dpms force off
335 exec --no-startup-id xset s off -dpms
336
337 ##### Brightness control
338 set $step 5%
339 bindsym XF86MonBrightnessUp exec --no-startup-id brightnessctl set +$step
340 bindsym XF86MonBrightnessDown exec --no-startup-id brightnessctl set $step-
341 exec_always --no-startup-id brightnessctl set 40%
342
343 #exec_always --no-startup-id swayidle \
344 #     timeout 5 'bash -c "brightnessctl set 30% && notify-send -u low
345 ↪ Idle: dimmed to 30%"' \
346 #     timeout 420 'bash -c "~/i3/bin/i3-lock.sh"' \
347 #     timeout 900 'bash -c "notify-send -u critical \"Suspending in 10s\"
348 ↪ && sleep 10 && systemctl suspend"' \
349 #     resume 'bash -c "brightnessctl set 100% && notify-send -u low
350 ↪ Resumed: brightness restored"' \
351 #     before-sleep '~/i3/bin/i3-lock.sh'
352 exec_always --no-startup-id /usr/bin/update-xidlehook-power-profile.sh
353
354 ##### doesn't cause issues like coordinate transformation
355 ↪ matrix, instead it just disables acceleration
356 #exec_always --no-startup-id xinput set-prop 'pointer:Razer Razer Viper Ultimate
357 ↪ Dongle' 'libinput Accel Profile Enabled' 0 1 0
358 #exec_always --no-startup-id xinput set-prop 'pointer:Razer Razer Viper Ultimate'
359 ↪ 'libinput Accel Profile Enabled' 0 1 0
360
361 ##### MODE SETTINGS #####
362
363 ##### simple menu
364 mode "(l)ock (s)leep (h)ybernate (r)eboot (p)oweroff (e)xit-i3" {
365     bindsym l exec ~/.i3/bin/i3-lock.sh; mode "default"
366     bindsym s exec ~/.i3/bin/i3-lock.sh && sleep 0.5 && systemctl suspend; mode
367     ↪ "default"
368     bindsym h exec ~/.i3/bin/i3-lock.sh && sleep 0.5 && systemctl hibernate;
369     ↪ mode "default"
370     bindsym p exec systemctl poweroff; mode "default"
371     bindsym r exec systemctl reboot; mode "default"
372     bindsym e exec --no-startup-id i3-msg exit; mode "default"
373     bindsym Return mode "default"
374     bindsym Escape mode "default"
375 }
376 bindsym $mod+Shift+e mode "(l)ock (s)leep (h)ybernate (r)eboot (p)oweroff
377 ↪ (e)xit-i3"
378
379 ##### resize windows
380 mode "resize" {
381     bindsym h resize shrink width 5 px or 5 ppt
382     bindsym j resize grow height 5 px or 5 ppt
383     bindsym k resize shrink height 5 px or 5 ppt
384     bindsym l resize grow width 5 px or 5 ppt
385 }

```

```

376
377     #greater resize
378     bindsym Shift+h resize shrink width 10 px or 10 ppt
379     bindsym Shift+j resize grow height 10 px or 10 ppt
380     bindsym Shift+k resize shrink height 10 px or 10 ppt
381     bindsym Shift+l resize grow width 10 px or 10 ppt
382
383     #back to normal: Enter or Escape
384     bindsym Return mode "default"
385     bindsym Escape mode "default"
386     bindsym $mod+r mode "default"
387 }
388 bindsym $mod+r mode "resize"
389
390 ##### gaps mode, isn't used a lot? could be removed
391 bindsym $mod+g mode "$mode_gaps"
392 set $mode_gaps gaps: (o) outer, (i) inner
393 set $mode_ogaps ogaps: (+) increase, (-) decrease, (0) set 0, (d) default gap
394 set $mode_igaps igaps: (+) increase, (-) decrease, (0) set 0, (d) default gap
395
396 mode "$mode_gaps" {
397     bindsym o      mode "$mode_ogaps"
398     bindsym i      mode "$mode_igaps"
399     bindsym Return mode "default"
400     bindsym Escape mode "default"
401 }
402
403 mode "$mode_ogaps" {
404     bindsym plus gaps outer current plus 5
405     bindsym minus gaps outer current minus 5
406     bindsym 0 gaps outer current set 0
407     bindsym d gaps outer current set $defouter
408
409     bindsym Return mode "default"
410     bindsym Escape mode "default"
411 }
412
413 mode "$mode_igaps" {
414     bindsym plus gaps inner current plus 5
415     bindsym minus gaps inner current minus 5
416     bindsym 0 gaps inner current set 0
417     bindsym d gaps inner current set $definner
418
419     bindsym Return mode "default"
420     bindsym Escape mode "default"
421 }
422
423 ##### enable the bars
424
425 bindsym $mod+b bar mode toggle
426 bar {
427     i3bar_command /usr/bin/i3bar
428     status_command i3status -c ~/.i3/i3status.conf
429     font pango:Iosevka 10
430     tray_output primary
431     colors {
432         background $nord_background
433         statusline $nord_foreground

```

```

434         focused_workspace $nord3 $nord8 $nord1
435         active_workspace  $nord3 $nord8 $nord5
436         inactive_workspace $nord8 $nord1 $nord5
437         urgent_workspace  $nord15 $nord1 $nord5
438     }
439 }
440 }
441
442 #bindsym $mod+b exec bash -c '~/config/polybar/launch.sh'
443
444 #####
445 ##### FOCUS COMMANDS #####
446 #####
447
448 ##### change focus
449 bindsym $mod+h focus left
450 bindsym $mod+j focus down
451 bindsym $mod+k focus up
452 bindsym $mod+l focus right
453
454 ##### move focused window
455 bindsym $mod+Shift+h move left
456 bindsym $mod+Shift+j move down
457 bindsym $mod+Shift+k move up
458 bindsym $mod+Shift+l move right
459
460 ##### change focus between tiling / floating windows
461 bindsym $mod+space focus mode_toggle
462
463 ##### focus the parent container
464 bindsym $mod+a focus parent
465
466 ##### focus the child container
467 bindsym $mod+z focus child
468
469 ##### focus urgent lasted $mod+'='
470 bindcode $mod+21 [urgent=latest] focus
471
472 ##### focus follow mouse
473 focus_follows_mouse no
474
475
476 #####
477 ##### LAYOUT COMMANDS #####
478 #####
479
480 ##### split in horizontal orientation
481 bindsym $mod+Shift+v split h
482
483 ##### split in vertical orientation
484 bindsym $mod+v split v
485
486 ##### enter fullscreen mode for the focused container
487 bindsym $mod+f fullscreen
488
489 ##### change container layout (stacked, tabbed, toggle split)
490 bindsym $mod+s layout stacking
491 bindsym $mod+w layout tabbed

```

```

492 bindsym $mod+e layout toggle split
493
494 ##### toggle tiling / floating
495 bindsym $mod+Shift+space floating toggle
496
497 ##### scratchpad
498 bindsym $mod+minus scratchpad show
499 bindsym $mod+Shift+minus move scratchpad
500
501
502 #####
503 #####
504 ##### PROGRAM SETTINGS #####
505 #####
506 #####
507
508 #####
509 ##### STARTUP PROGRAMS #####
510 #####
511
512 #####other
513 exec --no-startup-id nm-applet
514 exec --no-startup-id dex --autostart --environment i3
515 #exec --no-startup-id xss-lock --transfer-sleep-lock -- i3lock --nofork
516 exec --no-startup-id ibus-daemon --xim -d -r
517 #exec --no-startup-id ~/.config/polybar/launch.sh --cuts
518 #exec --no-startup-id ~/.i3/bin/bar
519 #exec --no-startup-id dunst
520
521
522 #####wallpaper/
523 #####headset/
524 #####bloatware
525 exec alacritty
526 exec sh -c 'LD_PRELOAD=/usr/local/lib/spotify-adblock.so spotify-launcher'
527 #exec elecwhat
528
529 ##### this is no longer used here, it should be used if spotify
530 ↳ doesn't throw an error at start
531 #exec --no-startup-id sleep 0.4 && ~/.i3/bin/cava-readjust.sh $tag0
532 #exec --no-startup-id caffeine start
533
534 ##### OTHER PROGRAMS #####
535 #####
536
537 ##### start a terminal
538 bindsym $mod+Return exec alacritty
539
540 ##### reminder
541 bindsym $mod+apostrophe exec i3-input -F "exec sh -c 'sleep %s && notify-send
542 ↳ Remember-%ss-ago!' -P 'Remind after: ' -P 'Title: '"
543
544 ##### nvim
545 bindsym $mod+$alt_mod+n exec alacritty -e nvim
546 bindsym $mod+$alt_mod+Shift+n exec alacritty -e vim
547
548 ##### htop y nvtop

```

```

548 bindsym $mod+$alt_mod+h exec alacritty --title htop -e htop
549 bindsym $mod+$alt_mod+Ctrl+h exec alacritty --title nvtop -e nvtop
550
551 ##### explore
552 bindsym $mod+$alt_mod+r exec alacritty -e ranger
553
554 ##### browser
555 bindsym $mod+$alt_mod+f exec firefox
556 bindsym $mod+$alt_mod+Shift+f exec firefox --private-window
557
558 ##### Spotify
559 bindsym $mod+$alt_mod+s exec sh -c 'LD_PRELOAD=/usr/local/lib/spotify-adblock.so
↪  spotify-launcher'
560 bindsym $mod+$alt_mod+Shift+s exec ~/.i3/bin/cava-reajust.sh $tag0
561
562 ##### clipmenu
563 #bindsym $alt_mod+c exec --no-startup-id alacritty --option window.opacity=0.5
↪  --title clipboard -e bash -c 'CM_LAUNCHER=fzf clipmenu -i'

```

## B. Configuración rofi de ejemplo

```
1  /**
2   * Nordic rofi theme
3   * Adapted by undiabler <undiabler@gmail.com>
4   *
5   * Nord Color palette imported from https://www.nordtheme.com/
6   *
7   */
8
9
10 * {
11     nord0: #2e3440;
12     nord1: #3b4252;
13     nord2: #434c5e;
14     nord3: #4c566a;
15
16     nord4: #d8dee9;
17     nord5: #e5e9f0;
18     nord6: #eceff4;
19
20     nord7: #8fbcb8;
21     nord8: #88c0d0;
22     nord9: #81a1c1;
23     nord10: #5e81ac;
24     nord11: #bf616a;
25
26     nord12: #d08770;
27     nord13: #ebcb8b;
28     nord14: #a3be8c;
29     nord15: #b48ead;
30
31     foreground: @nord9;
32     backlight: #ccffedd;
33     background-color: transparent;
34
35     highlight: underline bold #eceff4;
36
37     transparent: rgba(46,52,64,0);
38 }
39
40 window {
41     location: center;
42     anchor: center;
43     transparency: "screenshot";
44     padding: 10px;
45     border: 0px;
46     border-radius: 6px;
47
48     background-color: @transparent;
49     spacing: 0;
50     children: [mainbox];
```



```

51     orientation: horizontal;
52 }
53
54 mainbox {
55     spacing: 0;
56     children: [ inputbar, message, listview ];
57 }
58
59 message {
60     color: @nord0;
61     padding: 5;
62     border-color: @foreground;
63     border: 0px 2px 2px 2px;
64     background-color: @nord7;
65 }
66
67 inputbar {
68     color: @nord6;
69     padding: 11px;
70     background-color: #3b4252;
71
72     border: 1px;
73     border-radius: 6px 6px 0px 0px;
74     border-color: @nord10;
75 }
76
77 entry, prompt, case-indicator {
78     text-font: inherit;
79     text-color: inherit;
80 }
81
82 prompt {
83     margin: 0px 1em 0em 0em ;
84 }
85
86 listview {
87     padding: 8px;
88     border-radius: 0px 0px 6px 6px;
89     border-color: @nord10;
90     border: 0px 1px 1px 1px;
91     background-color: rgba(46,52,64,0.9);
92     dynamic: false;
93 }
94
95 element {
96     padding: 3px;
97     vertical-align: 0.5;
98     border-radius: 4px;
99     background-color: transparent;
100     color: @foreground;
101     text-color: rgb(216, 222, 233);
102 }
103
104 element selected.normal {
105     background-color: @nord7;
106     text-color: #2e3440;
107 }
108

```

```

109 element-text, element-icon {
110     background-color: inherit;
111     text-color:      inherit;
112 }
113
114 button {
115     padding: 6px;
116     color: @foreground;
117     horizontal-align: 0.5;
118
119     border: 2px 0px 2px 2px;
120     border-radius: 4px 0px 0px 4px;
121     border-color: @foreground;
122 }
123
124 button selected normal {
125     border: 2px 0px 2px 2px;
126     border-color: @foreground;
127 }
128
129 textbox {
130     padding: 8px;
131     border-radius: 6px 6px 6px 6px;
132     border-color: @nord10;
133     border: 1px 1px 1px 1px;
134     background-color: rgba(46,52,64,0.9);
135     dynamic: true;
136     text-color: @backlight;
137 }

```

# Bibliografía

- [1] Wikipedia. *GNU/Linux*. 2026. URL: <https://en.wikipedia.org/wiki/Linux> (visitado 24-02-2026).
- [2] i3. *i3wm*. 2026. URL: <https://i3wm.org/> (visitado 24-02-2026).
- [3] sway. *swaywm*. 2026. URL: <https://swaywm.org/> (visitado 24-02-2026).
- [4] Wikipedia. *Package\_manager*. Definición general de gestor de paquetes y su historia. 2026. URL: [https://en.wikipedia.org/wiki/Package\\_manager](https://en.wikipedia.org/wiki/Package_manager) (visitado 24-02-2026).
- [5] Arch Linux Wiki. *Pacman*. Pacman es el gestor de paquetes utilizado por Arch Linux, dicho gestor está escrito en C y busca ser minimalista y rápido. 2026. URL: <https://wiki.archlinux.org/title/Pacman> (visitado 24-02-2026).
- [6] Wikipedia. *Computer\_terminal*. 2026. URL: [https://en.wikipedia.org/wiki/Computer\\_terminal#Emulation](https://en.wikipedia.org/wiki/Computer_terminal#Emulation) (visitado 25-02-2026).
- [7] Arch Linux Wiki. *Install\_essential\_packages*. 2026. URL: [https://wiki.archlinux.org/title/Installation\\_guide#Install\\_essential\\_packages](https://wiki.archlinux.org/title/Installation_guide#Install_essential_packages) (visitado 25-02-2026).
- [8] Arch Linux Wiki. *Polkit*. 2026. URL: <https://wiki.archlinux.org/title/Polkit> (visitado 28-02-2026).
- [9] Arch Linux Wiki. *Reflector*. 2026. URL: <https://wiki.archlinux.org/title/Reflector> (visitado 28-02-2026).
- [10] Arch Linux Wiki. *iwd*. 2026. URL: <https://wiki.archlinux.org/title/Iwd> (visitado 07-03-2026).
- [11] Wikipedia. *X Window System*. 2026. URL: [https://en.wikipedia.org/wiki/X\\_Window\\_System](https://en.wikipedia.org/wiki/X_Window_System) (visitado 15-03-2026).
- [12] Wikipedia. *Wayland (protocol)*. 2026. URL: [https://en.wikipedia.org/wiki/Wayland\\_\(protocol\)](https://en.wikipedia.org/wiki/Wayland_(protocol)) (visitado 15-03-2026).
- [13] i3. *i3wm configuration docs*. 2026. URL: <https://i3wm.org/docs/userguide.html> (visitado 14-04-2026).
- [14] Wikipedia. *Compositing\_manager*. Definición general de lo que es un compositor. 2026. URL: [https://en.wikipedia.org/wiki/Compositing\\_manager](https://en.wikipedia.org/wiki/Compositing_manager) (visitado 14-04-2026).
- [15] Arch Linux Wiki. *picom*. 2026. URL: <https://wiki.archlinux.org/title/Picom> (visitado 14-04-2026).
- [16] Yuxuan Shui. *picom configuration*. 2026. URL: <https://picom.app/> (visitado 14-04-2026).
- [17] Arch Linux Wiki. *rofi*. 2026. URL: <https://wiki.archlinux.org/title/Rofi> (visitado 14-04-2026).
- [18] davatorium. *rofi docs*. 2026. URL: <https://davatorium.github.io/rofi/> (visitado 14-04-2026).
- [19] Arch Linux Wiki. *fonts*. 2026. URL: <https://wiki.archlinux.org/title/Fonts> (visitado 15-04-2026).

- [20] Arch Linux Wiki. *icons*. 2026. URL: <https://wiki.archlinux.org/title/Icons> (visitado 15-04-2026).
- [21] Arch Linux Wiki. *pipewire*. 2026. URL: <https://wiki.archlinux.org/title/Polkit> (visitado 15-04-2026).
- [22] Wikipedia. *Advanced\_Linux\_Sound\_Architecture*. 2026. URL: [https://es.wikipedia.org/wiki/Advanced\\_Linux\\_Sound\\_Architecture](https://es.wikipedia.org/wiki/Advanced_Linux_Sound_Architecture) (visitado 15-04-2026).
- [23] Arch Linux Wiki. *xinit*. 2026. URL: <https://wiki.archlinux.org/title/Xinit> (visitado 16-04-2026).
- [24] Arch Linux Wiki. *xinit*. 2026. URL: <https://wiki.archlinux.org/title/Systemd/Journal> (visitado 16-04-2026).