

---

# **Xfce docs**

***Release 4.16***

**Xfce developers**

**Mar 20, 2022**



## GETTING STARTED:

<b>1</b>	<b>Getting Started with Xfce</b>	<b>3</b>
1.1	Introduction . . . . .	3
1.2	Installation . . . . .	4
1.3	Running Xfce . . . . .	4
1.4	The Desktop Environment . . . . .	5
1.5	Common Tasks . . . . .	5
1.6	Further Reading . . . . .	6
<b>2</b>	<b>Xfce - Frequently Asked Questions</b>	<b>7</b>
2.1	General Questions . . . . .	9
2.2	Keyboard Related . . . . .	10
2.3	Application Menu . . . . .	13
2.4	Desktop Manager . . . . .	14
2.5	Panel . . . . .	14
2.6	Themes . . . . .	15
2.7	Miscellaneous . . . . .	15
<b>3</b>	<b>Building Xfce from source</b>	<b>19</b>
3.1	Choosing your Xfce version . . . . .	20
3.2	Obtaining the Xfce source code . . . . .	20
3.3	Xfce Dependencies explained . . . . .	21
3.4	Building the packages . . . . .	23
3.5	Finding additional Help . . . . .	24
<b>4</b>	<b>How to contribute to Xfce</b>	<b>25</b>
4.1	Bug Reporting and Testing . . . . .	25
4.2	Communication . . . . .	26
4.3	Development . . . . .	26
4.4	Documentation . . . . .	26
4.5	Financial Contributions . . . . .	26
4.6	Feature Requests . . . . .	27
4.7	Translation . . . . .	27
<b>5</b>	<b>Bug Reporting and Fixing</b>	<b>29</b>
5.1	Crashes . . . . .	29
5.2	Search First . . . . .	30
5.3	Report . . . . .	30
5.4	Backtraces . . . . .	30
5.5	Fixing Bugs . . . . .	31
<b>6</b>	<b>Core Modules</b>	<b>33</b>

6.1	xfce4-appfinder - Application Finder . . . . .	33
6.2	xfconf - Configuration Storage System . . . . .	35
6.3	xfdesktop - Xfce Desktop Manager . . . . .	37
6.4	xfce4-dev-tools . . . . .	39
6.5	Exo - Helper Applications . . . . .	41
6.6	Thunar File Manager . . . . .	43
6.7	garcon - menu library . . . . .	46
6.8	xfce4-panel - Xfce Panel . . . . .	48
6.9	xfce4-power-manager - Power Manager . . . . .	51
6.10	xfce4-session - Session Manager . . . . .	53
6.11	xfce4-settings - Settings Manager . . . . .	54
6.12	Tumbler - Thumbnail Service . . . . .	56
6.13	libxfce4util - Utility sharing library . . . . .	59
6.14	libxfce4ui - widget sharing library . . . . .	60
6.15	xfwm4 - Window Manager . . . . .	61
<b>7</b>	<b>Applications</b>	<b>65</b>
<b>8</b>	<b>Getting Started</b>	<b>67</b>
<b>9</b>	<b>Documentation for Xfce Applications</b>	<b>69</b>
9.1	Core Modules . . . . .	69
9.2	Applications . . . . .	69

This website provides user documentation for the latest stable version of the Xfce desktop environment. Go to [Xfce 4.14](#) for the previous version.



## GETTING STARTED WITH XFCE

### Contents

- *Getting Started with Xfce*
  - *Introduction*
    - \* *System Requirements*
  - *Installation*
  - *Running Xfce*
  - *The Desktop Environment*
  - *Common Tasks*
    - \* *Running Programs*
    - \* *Managing Windows and Workspaces*
    - \* *Using the Settings Manager*
  - *Further Reading*

## 1.1 Introduction

Xfce is a lightweight desktop environment for Unix-like operating systems. It aims to be fast and light on system resources, while still being visually appealing and easy to use.

Xfce embodies the traditional UNIX philosophy of modularity and re-usability. It consists of a number of components that provide the full functionality one can expect of a modern desktop environment. They are packaged separately and you can pick among the available packages to create the optimal personal working environment.

Another priority of Xfce is an adherence to standards, specifically those defined at freedesktop.org.

### 1.1.1 System Requirements

Even though Xfce is very light-weight, there are some distribution specific minimum requirements.

## 1.2 Installation

The Xfce project officially only releases source code for the desktop environment. However, binary packages may have been contributed by other people for your OS or distribution. Here some links to distribution specific installation instructions:

- Arch
- Debian
- Fedora
- FreeBSD
- Gentoo
- Mageia
- MidnightBSD ( or possibly `pkg_add -r xfce4 ?` )
- Mythbuntu: `sudo apt-get install mythbuntu-desktop`
- NetBSD (`pkg_add xfce4`)
- Ubuntu: `sudo apt-get install xubuntu-desktop` (package description; direct install)

Detailed instructions on compiling Xfce yourself can be found [here](#).

## 1.3 Running Xfce

- Display Managers

Xfce4-session installs a file that should add an option for display managers to run an Xfce session. The Xfce Desktop Environment does not have its own DM, but various options are available like gdm, slim, lxdm and lightdm. Check [this link](#) for details.

- Command line

Use `startxfce4` to start an Xfce session or choose Xfce Session from the login manager, which includes the session manager, the panel, the window manager and the desktop manager. See [auto login from console](#) for more information.

By default, the Xfce session manager manages the startup of applications. It allows you to save your session when you quit Xfce, so that the next time you log in, the same applications will be started for you automatically.



## 1.4 The Desktop Environment

The Xfce Desktop Environment is not a single entity that provides all functionality, but rather it tries to adhere to the old UNIX tradition of small tools that do one job and do it well.

Default Xfce session, with desktop manager and panel

When you start the Xfce session for the first time, several applications are started by the Xfce session manager:

- **Panel** In a default session there is a full width panel at the top of the screen and a smaller one at the bottom. The Panel application manages all panels on the screen. The top panel shows a graphical pager with a miniature view of all workspaces, a task list showing all applications running on the current workspace and a system tray to show status icons that are used for example by some media players or instant messaging applications. The bottom panel contains several application launchers and a clock. You can right-click on any panel item to get a menu that allows you to change its properties, add or remove new items or to change the properties of the panel itself.
- **Desktop Manager** The desktop manager provides the desktop background image and two menus when you click on the desktop background. Optionally, it can show icons on the desktop, either for minimized applications or for files in the ~/Desktop folder. The right mouse button opens a menu that allows you to start applications. Look at the manual to find out how to change the menu contents. The middle mouse button (or Shift + left click) opens a list of all applications that are currently running. You can activate an application by clicking on its menu entry.
- **Window Manager** The Window Manager is responsible for placing the windows on the screen and provides the window borders and decorations. It allows you to move windows around by dragging the title bar, and provides title bar buttons: for example to close, minimize or maximize a window. Look at the manual for a detailed explanation of the window manager settings.
- **Settings Manager** The settings manager runs in the background and ensures that all Xfce applications update their settings when the user changes something within the application. It is also responsible for reading the configuration files on disk at startup. Look at the manual for a detailed explanation of the settings manager.

## 1.5 Common Tasks

This section will explain how to perform several common tasks to quickly get you started working with Xfce and familiar with its applications.

### 1.5.1 Running Programs

- **Xfce Panel** The panel can be used to allow quick access to your most frequently used applications by means of launchers, these application launchers are displayed as icons on your panel that launch your specified program. The Applications Menu on the panel also contains all installed programs.
- **Desktop Menu** Another method for starting applications is from the desktop right-click menu. Read the Desktop Manager manual for more information on how to change the menu contents.
- **Application Finder** If you know the name of the application you want to launch you can use the Application Finder. To open the dialog, press Alt-F2 or choose the Run Program... option from the desktop menu.

## 1.5.2 Managing Windows and Workspaces

- **Basic window operations** You can move a window around the screen by dragging on its title bar. A window can be closed, minimized, maximized, shaded and made sticky (make it show up on all workspaces) by using the title bar buttons. Right-clicking on the title bar will open a menu that gives access to all window operations. Shading a window, or rolling it up to hide its contents and only show the title bar, can also be accomplished by scrolling the mouse wheel while hovering over the title bar. Mouse wheel up is shade, mouse wheel down is unshade. If you don't want maximized windows to cover up the entire screen, you can set workspace margins in the Settings Manager.
- **Application management** To see what applications are currently running, you can look at the task list on the panel. Clicking on an application in the task list will focus it, and clicking it again will hide it. When you click with the middle mouse button on the desktop background a list of windows is shown, ordered by workspace. You can activate the application or change workspaces by choosing the appropriate menu entry.
- **Workspaces** You can change workspaces by clicking on them in the graphical pager on the panel. Pressing Ctrl-Alt-LeftArrow or Ctrl-Alt-RightArrow will cycle through the workspaces. Scrolling the mouse wheel while hovering over the pager or the desktop background has the same effect. To add or remove workspaces, you can use the middle-click desktop menu or the settings dialog (see below).

## 1.5.3 Using the Settings Manager

The Settings Manager provides access to the global preferences of many aspects of the Xfce Desktop Environment. You can run it by pressing its launcher on the panel, from the desktop mouse menu or by running `xfce4-settings-manager`.

Refer to the individual manuals of the Xfce components for more information on settings specifically for that application. Don't be afraid to explore all the settings to find out what options are available that allow you to create the best possible working environment.

## 1.6 Further Reading

This guide provides only a general overview of the Xfce Desktop Environment. More information is available in the manuals of the individual components of Xfce.

## XFCE - FREQUENTLY ASKED QUESTIONS

### Contents

- *Xfce - Frequently Asked Questions*
  - *General Questions*
    - \* *How to pronounce Xfce ?*
    - \* *What does it mean ?*
    - \* *What does the logo mean ?*
    - \* *On which platforms does Xfce run currently?*
    - \* *Under which license is Xfce distributed?*
    - \* *How long between two official releases?*
    - \* *Where can I find Xfce information/support for my specific distribution?*
    - \* *Distribution Specific Xfce Info/Support Pages:*
    - \* *General Xfce forum*
  - *Keyboard Related*
    - \* *How to configure Shortcuts / Hotkeys / Menu Accelerators ?*
    - \* *Is there some way to call the menu with the keyboard in Xfce?*
    - \* *Is it possible to focus the Verve plugin with a key?*
    - \* *My windows button does not work in the Keyboard Settings > Shortcuts.*
    - \* *How do I get numlock to start on login?*
    - \* *Is it possible to use Media keys in the Shortcut Editor?*
    - \* *How to determine keycodes with “xev”*
    - \* *What should I do to change keyboard layout?*
    - \* *Is it possible to change the default shortcut keys?*
    - \* *How can I see a list of all the shortcut keys?*
    - \* *How do I make a shortcut that doesn’t steal focus?*
  - *Application Menu*
    - \* *The left-click to get the menu on the title bar menu button seems a little slow. How do I change that?*

- \* *How do I display a list of all windows?*
- \* *How to edit the auto generated menu with the menu editor?*
- \* *What are the exact commands used when launching the ‘Settings’ applications?*
- *Desktop Manager*
  - \* *I want to disable the trash, home and filesystem icons on my desktop, is that possible?*
  - \* *My Xfce Desktop doesn’t have any shortcut icons, why?*
- *Panel*
  - \* *Is it possible to change the icon used by the icon box or task bar for a given application?*
  - \* *I’ve installed a plug-in for the panel, but the indicators don’t use different colors. What can I do?*
  - \* *How do I set the panel layer in Xfce 4.2 and 4.4?*
  - \* *Is there a world clock applet?*
- *Themes*
  - \* *The gtk-xfce-engine themes do not appear in the “user interface” settings dialog*
  - \* *How can I customize my Xfce desktop environment*
  - \* *Where can I find additional backgrounds and wallpapers for my desktop?*
- *Miscellaneous*
  - \* *Why the file chooser is very slow ?*
  - \* *Why the response to Terminal application is slow?*
  - \* *How do I enable panel transparency and window shadows?*
    - *ATI users (X.org radeon driver)*
    - *nVidia users*
  - \* *Kiosk - I want to use Xfce in a corporate environment and not let users modify their menu.*
  - \* *How to use a different Window Manager*
  - \* *How to use SCIM in Xfce*
  - \* *On Login: When I start Xfce a dialog pops up saying “Could not look up internet address for...”*
  - \* *On Logout: When I try to log out by pressing the logout button in the panel, I get a dialog asking me whether I want to quit the panel and/or xfce4-session-logout reports that no session manager is running, but it is!*
  - \* *How can I install a new font in Xfce?*

## 2.1 General Questions

### 2.1.1 How to pronounce Xfce ?

“Ecks Eff See Eee”

### 2.1.2 What does it mean ?

The name Xfce originally stood for XForms Common Environment, but since then, Xfce was rewritten twice and doesn't use XForms toolkit anymore. The name survived, but the F is no longer capitalized (not “XFce”, but “Xfce”). Currently the abbreviation doesn't stand for anything (suggestion: X Freakin' Cool Environment). It's not pronounced “X-Face”. There is no “a” in it.

### 2.1.3 What does the logo mean ?

A mouse, obviously, for all kinds of reasons like world domination and monsters and such.

### 2.1.4 On which platforms does Xfce run currently?

Xfce is developed to be versatile. It is currently supported on Linux, Solaris and BSD, but has been known to run in some shape or form on IRIX, MacOS X, and Windows.

### 2.1.5 Under which license is Xfce distributed?

Xfce 4 components are licensed under free or open source licenses: GPL or BSDL for applications and LGPL or BSDL for libraries. Read the documentation, the source code, or the Xfce homepage for more information.

### 2.1.6 How long between two official releases?

There is no set schedule, but there are goals the developers try to meet. That said, the creation of deadlines does not lend itself well to those working without compensation. So the overall goal is to release a new version as certain goals are reached. Unfortunately, that does not allow the advanced statement of any release schedule. Please check back often to read any news releases about the product.

### 2.1.7 Where can I find Xfce information/support for my specific distribution?

If you are not able to find support on the Xfce Forums, you may want to use a site focused, specifically, on the distribution you are currently running. The following is not an exhaustive list of Xfce resources.

## 2.1.8 Distribution Specific Xfce Info/Support Pages:

- ArchWiki
- Debian Xfce
- Fedora Xfce Desktop
- **FreeBSD Xfce**
  - FreeBSD Xfce FAQ
- Manjaro Xfce
- Linux Mint Xfce Forums
- **Xubuntu**
  - Ubuntuforums
  - AskUbuntu

## 2.1.9 General Xfce forum

- Xfce on Reddit
- 

## 2.2 Keyboard Related

### 2.2.1 How to configure Shortcuts / Hotkeys / Menu Accelerators ?

A number of Xfce applications (Thunar, for example) support the standard GTK2 way of changing shortcuts: simply hover over the menu option with the mouse pointer and press the keyboard shortcut you want to rebind it to.

To delete a keyboard assignment, press the **Backspace** key while you are on the menu entry.

If the shortcut doesn't change, then you need to enable the feature in GTK+. This can be achieved in 3 ways:

- If you are running the Xfce desktop environment, enable **Editable menu accelerators** in the User Interface Preferences dialog.
- If you are running GNOME then you can enable **Editable menu accelerators** in the **Menu and Toolbars** control center dialog.
- Otherwise put the following in your `~/.gtkrc-2.0` file (create the file if it doesn't exist):  
`gtk-can-change-accel=1`

**Warning:** When `xfsettingsd` is running you must change the setting with the Xfce GUI, not through the `.gtkrc-2.0` file.

**Caution:** This functionality has been disabled since GTK3 which means that Xfce apps that have migrated to GTK3 (such as `xfce4-terminal`) do not support it. Refer to specific app's documentation to learn how to configure its shortcuts.

## 2.2.2 Is there some way to call the menu with the keyboard in Xfce?

Assign a key with the Keyboard Settings > Shortcuts to the command `xfdesktop -menu`. (This does not work reliably since Linux Kernel is tickless, so `xfdesktop -menu` needs a fix) The menu will popup where your mouse is located. You can also use `xfce4-popup-applicationsmenu` to popup the panel menu (also provided by `xfdesktop` and make sure you have the plugin in your panel `^_~`).

## 2.2.3 Is it possible to focus the Verve plugin with a key?

Assign a key to the command `verve-focus`

## 2.2.4 My windows button does not work in the Keyboard Settings > Shortcuts.

The windows button (also known as the superkey) not working as a modifier is related to the toolkit, GTK+ in the case of Xfce. If you want to have the windows-key working we recommend you to upgrade GTK+ to at least version 2.10.0.

## 2.2.5 How do I get numlock to start on login?

There are two possibilities to achieve this. Or you should use a display manager that turns the numlock on (eg. `gdm`, check the settings) or you can use a little program called `numlockx`, adding `numlockx` on in your `.xinitrc` will do the job.

## 2.2.6 Is it possible to use Media keys in the Shortcut Editor?

Use `xmodmap` to assign keycodes to your Media keys to make them available for the Xfce shortcut editor:

To determine keycodes of the multimedia keys use the program `xev`. Create a `.Xmodmap` file in your `$HOME` directory containing those keycodes and assign keysyms to them.

Here are some examples of keycodes and their respective keysyms:

```
keycode 162 = XF86AudioPlay
keycode 164 = XF86AudioStop
keycode 160 = XF86AudioMute
keycode 144 = XF86AudioPrev
keycode 153 = XF86AudioNext
keycode 176 = XF86AudioRaiseVolume
keycode 174 = XF86AudioLowerVolume
keycode 237 = XF86AudioMedia
keycode 230 = XF86Favorites
keycode 236 = XF86Mail
keycode 178 = XF86WWW
```

All possible keysyms can be found in `/usr/lib/X11/XKeysymDB` or `/usr/share/X11/XKeysymDB`. To ensure that the `.Xmodmap` file is loaded when you start Xfce add `/usr/bin/xmodmap $HOME/.Xmodmap` to your `.xinitrc` or `.xprofile` file. When you start the shortcut editor, the assigned keysyms should show up when you press one of your multimedia keys. Now it is possible to assign a command to them.

**Note:** Several problems with auto-loading of `.Xmodmap` files at Xfce startup have been reported (also when issued

as autostart command). Search the Xfce Bugzilla sites for current problems. As a workaround, run `xmodmap ~/.Xmodmap` by hand every time, or try out editing the somewhat less straightforward `xkb` configuration files.

---

## 2.2.7 How to determine keycodes with “xev”

All keyboards are different, keycodes can differ and not everyone has time to search XKeysymDB file. You can acquire keycodes manually from your keyboard using the application `xev`.

In a terminal type the following:

```
xev | grep -A2 --line-buffered '^KeyRelease' | sed -n 's/^.*keycode \([0-9]*\).*  
↪ * (. *, \(. *\)).*$/\1 \2/p'
```

Next, press the key that you need the keycode from (e.g. When the key “Stop” is pressed, the output is “174 XF86AudioStop”).

## 2.2.8 What should I do to change keyboard layout?

There are several options. One is to use `xfce4-xkb-plugin`, see `xkb` plugin . You can also use the `setxkbmap` command with the two letter keyboard code as argument; you can edit your X server configuration file to choose a different keyboard layout (change the value after Option “`XkbLayout`”, e.g.: Option “`XkbLayout`” “`dvorak`”).

## 2.2.9 Is it possible to change the default shortcut keys?

Yes, of course. Keyboard shortcuts are defined in two locations. The shortcuts to handle the window manager are defined in the Settings Manager > Window Manager Settings > Keyboard. The Default theme can not be changed; but, when you add a theme you can change that the theme you just added.

More global keyboard shortcuts, like volume adjustments, can be found in Settings Manager > Keyboard Preferences > Shortcuts. Again, you need to add a new theme before you can start customizing it.

## 2.2.10 How can I see a list of all the shortcut keys?

Use the following command, which will produce a nicely formatted text list to standard output:

```
xfconf-query -c xfce4-keyboard-shortcuts -l -v | cut -d '/' -f4 | awk '{printf "%30s", $2;  
↪ print "\t" $1}' | sort | uniq
```

If you want to put this list into a file, add `> filename` at the end of command.

## 2.2.11 How do I make a shortcut that doesn’t steal focus?

It is not currently possible to do this.

Back To Top

---



## 2.3 Application Menu

### 2.3.1 The left-click to get the menu on the title bar menu button seems a little slow. How do I change that?

The left-button single-click menu button display speed is linked to the double click speed. If one wants the menu to appear quicker, just change the double click speed in the Xfce 4 Settings Manager Mouse properties to be faster. Or, one can right click on the title bar to get the menu displayed almost instantly without adjusting the double-click speed. The menu will display both ways.

### 2.3.2 How do I display a list of all windows?

There are two possibilities. The first is by middle clicking on the desktop (if you have `xfdesktop` running) or you can add the window list plugin to the panel (is provided with a `xfce4-popup-windowlist` command).

### 2.3.3 How to edit the auto generated menu with the menu editor?

```
cp ~/.cache/xfce4/desktop/menu-cache-name-of-the-generated-file.xml ~/.config/xfce4/
↳desktop/menu2.xml
cd ~/.config/xfce4/desktop/
cat menu.xml > menu3.xml
cat menu2.xml >> menu3.xml
mv menu.xml menu.orig.xml
mv menu3.xml menu.xml
```

Now, you already have a menu with all the categories in the main tree with some duplicates, but you must first edit `menu.xml` with your favorite editor and remove the 4 following lines in the middle of the file, otherwise the menu editor will complain about a wrong format:

```
</xfdesktop-menu>
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE xfdesktop-menu>

<xfdesktop-menu>
```

That's all. Now you can run the menu editor, remove the few duplicates and edit all as you like.

Settings > Desktop > Menu > Menu Editor

**Note:** by removing the “system” line, you will remove all the duplicates menu entries from the auto generated file. So, if it is changed in this auto generated file, they don't appear anymore, but you will get rid of most of the duplicates.

To restore the original menu, just do in a terminal:

```
mv menu.xml menu3.xml; mv menu.orig.xml menu.xml
```

### **2.3.4 What are the exact commands used when launching the ‘Settings’ applications?**

Please see this wiki entry for a list of the precise commands run for each entry under the ‘Settings’ menu in a default installation of Xfce4.

[Back To Top](#)

---

## **2.4 Desktop Manager**

### **2.4.1 I want to disable the trash, home and filesystem icons on my desktop, is that possible?**

Yes. In Settings > Desktop > Icons > Appearance, select ‘None’.

### **2.4.2 My Xfce Desktop doesn’t have any shortcut icons, why?**

You can adjust this via Settings > Desktop > Icons > Appearance.

[Back To Top](#)

---

## **2.5 Panel**

### **2.5.1 Is it possible to change the icon used by the icon box or task bar for a given application?**

It’s not possible. This setting has to be managed by the application itself.

### **2.5.2 I’ve installed a plug-in for the panel, but the indicators don’t use different colors. What can I do?**

First, try another Gtk theme, since some themes override the color. If it doesn’t solve the problem, you probably have an old `~/gtkrc-2.0` : remove it and try again.

### **2.5.3 How do I set the panel layer in Xfce 4.2 and 4.4?**

In order to improve focus management this option was removed.

### 2.5.4 Is there a world clock applet?

You need to add the Orage Clock to the panel. Then you can middle-click the clock to open the “Global Time” window, to which you can add any number of clocks.

[Back To Top](#)

---

## 2.6 Themes

### 2.6.1 The gtk-xfce-engine themes do not appear in the “user interface” settings dialog

The gtk-xfce-engine-2 package has to be installed using same prefix as Gtk2 itself. When installed from sources, the engine is, by default, installed in `/usr/local`, while Gtk2 is often installed in `/usr`. Just install `gtk-xfce-engine-2` again using `./configure --prefix=/usr`, and the themes will hopefully become available.

### 2.6.2 How can I customize my Xfce desktop environment

You can read everything about changing themes in the [How to install new themes](#) wiki page. See [Xfce Look](#) for a selection of themes geared towards Xfce/Xfwm4.

### 2.6.3 Where can I find additional backgrounds and wallpapers for my desktop?

Here is a selection of beautiful images and pointers to other sites with quality, high-resolution images.

[Back To Top](#)

---

## 2.7 Miscellaneous

### 2.7.1 Why the file chooser is very slow ?

It is more likely that the icon theme you are using renders too many SVGs making it very hard to scroll. Switch to another icon theme.

### 2.7.2 Why the response to Terminal application is slow?

For NVidia users, add this to your settings:

```
nvidia-settings -a InitialPixmapPlacement=0 -a GlyphCache=1
```

For all users, your driver may not support `argb` visuals very well. You can disable it for Terminal by exporting the environment variable `XLIB_SKIP_ARGB_VISUALS=1`. To disable it for Terminal only, put the next lines inside `~/bin/terminal` for example (given you have a personal `bin` directory, you can also put it inside `/usr/local/bin`):

```
#!/bin/sh
XLIB_SKIP_ARGB_VISUALS=1 /usr/bin/Terminal
```

### 2.7.3 How do I enable panel transparency and window shadows?

Enable the Composite extension in the X11 config file and make sure Xfwm4 is compiled with embedded compositor (xfwm4 -V).

```
Section "Extensions"
    Option "Composite" "Enable"
EndSection
```

Pay attention: recent versions of X.org turn composite on by default. If you experience speed problems or any other glitches you have to disable it explicitly:

```
Section "Extensions"
    Option "Composite" "Disable"
EndSection
```

If you have a reasonably new X.org (7.1, possibly 7.0) and your graphics card is listed as “supported” at X.org’s EXA status page, you should also enable EXA by adding this line to the card’s Device section in your xorg.conf:

```
Option "AccelMethod" "exa"
```

Enabling EXA will normally provide a speed increase for compositing and font rendering, but may cause a small reduction in OpenGL rendering speed.

Once the Composite extension is activated, go to **Settings > Panel and Settings > Window Manager Tweaks**.

#### ATI users (X.org radeon driver)

ATI R3xx/R4xx (9500 to X850, X1050) users may also need this in the device section for the card:

```
Option "MigrationHeuristic" "greedy"
Option "AccelDFS" "true"                # but see radeon(4)
Option "EnablePageFlip" "true"
Option "EnableDepthMoves" "true"
```

#### nVidia users

NVidia users may also need this in the device section for the card:

```
Option "RenderAccel" "true"
Option "AllowGLXWithComposite" "true"
```

Read /usr/share/doc/nvidia-glx/README.txt.gz (and search for “RenderAccel” and “AllowGLXWithComposite”) to see if they are recommended at all for your system. At least for recent NVidia GLX drivers, “AllowGLXWithComposite” “true” is only for X servers older than X11R6.9.0, and “RenderAccel” “true” is the default setting, and therefore not required. If you are running a recent NVidia driver and a recent xorg-server, you do not need these settings (and should not use the “AllowGLXWithComposite” “true” setting).

## 2.7.4 Kiosk - I want to use Xfce in a corporate environment and not let users modify their menu.

Use kiosk mode (see also `xfce4-session` docs).

## 2.7.5 How to use a different Window Manager

Please refer to this manual page for instructions and caveats regarding the running of a Window Manager other than `fvwm4`.

## 2.7.6 How to use SCIM in Xfce

Refer to this manual

## 2.7.7 On Login: When I start Xfce a dialog pops up saying “Could not look up internet address for...”

Xfce simply wants your hostname to be in `/etc/hosts`. Example input: `127.0.0.1 localhost`

## 2.7.8 On Logout: When I try to log out by pressing the logout button in the panel, I get a dialog asking me whether I want to quit the panel and/or `xfce4-session-logout` reports that no session manager is running, but it is!

For some reason, your X applications can not connect to the session manager.

**Possible causes are:**

- Your hostname cannot be resolved (see Login problems section).
- Your home partition or partition containing `/tmp` is filled up.
- Your hostname contains non-ascii characters (no umlauts allowed, in particular)
- Either `~/.ICEauthority` or `/tmp/.ICE-unix` has wrong permissions.

Also check `.xsession-errors` for clues.

## 2.7.9 How can I install a new font in Xfce?

Refer to your distribution-specific support as this is handled by your distro and not, specifically Xfce.



## BUILDING XFCE FROM SOURCE

To compile a complete Xfce desktop from source requires some information about the dependency chain and the various configure options available in each module. This page will describe this in detail, so you can compile your own Xfce desktop; both system wide as well as in your \$HOME directory.

### Contents

- *Building Xfce from source*
  - *Choosing your Xfce version*
  - *Obtaining the Xfce source code*
    - \* *Released packages*
    - \* *From the code repository*
  - *Xfce Dependencies explained*
    - \* *Build requirements*
    - \* *Xfce's dependency chain*
    - \* *Package specific dependencies*
  - *Building the packages*
    - \* *Installation location and variables*
    - \* *Compiling and installing*
    - \* *Understanding Debug levels*
      - *no*
      - *minimum*
      - *yes*
      - *full*
  - *Finding additional Help*

## 3.1 Choosing your Xfce version

First you will need to decide on which version of Xfce that you want to compile. Xfce version numbers are constructed like this: 1.2.3 => major.minor.micro. If the minor version is an even number, this means that it is a stable release. An odd numbered minor version indicate a development, or testing snapshots. The micro number increases with each release. Also, it is recommended to keep the major.minor version the same for the core components (some packages don't follow those numbers) or there will most likely be dependency version problems during compilation.

The latest releases available can be found on the [download](#) page. You can also build from the source repositories if you have enough experience. Additional information about this is provided below.

---

## 3.2 Obtaining the Xfce source code

Once you have decided which version that you want to compile, there are a couple of ways to download the source code. You can either choose to build official released packages, or clone the source repositories. Choosing the released packages is the recommended option. The source repositories could possibly be broken and are not generally suitable for production environments. However, they always contain the latest features, bug fixes and translations.

If you are not sure which to choose, it is easiest to begin with the latest stable fat-tarball of a collection release.

### 3.2.1 Released packages

Xfce packages have two types of releases: collection releases and individual releases. Click the links below to open the relevant download location.

- [Collection releases of the core modules](#) From time to time a new collection release of Xfce is made. A collection release consists of a number of individual tarballs and a fat-tarball that contains all of the individual tarballs. The collection releases always contains the correct versions needed to build the other packages in the release. If you want the latest version of each package, you can start with a collection release and then look for individual bug fix releases, explained in the item below.
- [Individual releases](#) Each package in Xfce is allowed to make development and stable releases at any time, see the [release model](#) documentation for more information.

### 3.2.2 From the code repository

For the latest code, you need to make local copies of the [GIT repositories](#). This is the place where developers submit their code. If you decide to use GIT, try to use the same branch for all the packages to avoid dependency version problems.

---

**Note:** For additional information about using Git with Xfce, see the [Xfce and git information](#) page.

---

You first need to make clones of each GIT repository, like the command below, where `$category` is for example `xfce` and `$module` `xfce4-panel`. If you click on a module in the [online repository](#) list, the clone uri is shown at the bottom of the page.

```
git clone https://gitlab.xfce.org/$category/$module
```



After the repository is cloned, you will need to choose which branch you want to compile. By default, the `master` branch is selected, but there are also branches for stable releases like `xfce-4.16`. The stable branches contain the latest bug fixes without any major new features, the master branch is the development version for the next release of Xfce. Both should not be used in a production environment.

To switch to another branch (`git branch -a` lists all available branches) run the following command inside the cloned directory to switch to, for example, the `xfce-4.16` branch:

```
git checkout --track origin/xfce-4.16
```

After this, you can update your local copy of the repository with `git pull` and switch to another branch with `git checkout $branchname`. Read the [GIT manuals](#) for more information about using GIT.

[Back To Top](#)

---

## 3.3 Xfce Dependencies explained

### 3.3.1 Build requirements

Before you can start compiling, you need a couple of packages that are required to compile Xfce. Use your distributions package manager to install those packages. Additional packages are required too for some modules, but we'll get to that later.

- [A working GNU toolchain](#)
- **Gtk+ and Glib headers, in some distributions called the -devel packages**
  - Xfce 4.14 requires Gtk+ 3.22 and Glib 2.50
  - pkgconfig
  - Coffee or tea

For history, here are the versions needed for older xfce releases, which are not supported anymore

- Xfce 4.16 requires Gtk+ 3.22 and Glib 2.42
- Xfce 4.12 requires Gtk+ 2.24 and Glib
- Xfce 4.10 requires Gtk+ 2.20 and Glib 2.24
- Xfce 4.8 requires Gtk+ 2.14 and Glib 2.20
- Xfce 4.6 requires Gtk+ 2.10 and Glib 2.12

### 3.3.2 Xfce's dependency chain

The Xfce packages need to be built in a specific order. If you don't follow this, compile options might not be available or the configure stage will abort because of missing dependencies.

- xfce4-dev-tools (only required if you build from GIT)
- libxfce4util
- xfconf
- libxfce4ui
- garcon, exo

- thunar
- xfce4-panel, xfce4-settings, xfce4-session, xfdesktop, xfwm4, xfce4-appfinder, tumbler...

### 3.3.3 Package specific dependencies

Some core packages in Xfce have additional or optional dependencies, all are listed in the table below for the latest stable version. We only show the top-dependency (so xfce4-panel also requires gtk+, but libxfce4ui already depends on that).

---

**Tip:** Note that some of the package-names may vary between distributions.

---

	Dependency(-ies)	Optional Dependency(-ies)
xfce4-dev-tools	autoconf, automake, intltool, pkgconfig	
libxfce4util	glib2	
xfconf	libxfce4util, gdbus	
libxfce4ui	libxfce4util, gtk+, xfconf, glibtop	libstartup-notification, libgladeui
garcon	gio, libxfce4util	
exo	libxfce4util, gtk+, perl-uri, libxfce4ui	
xfce4-panel	libxfce4ui, exo, garcon, libwnck	libstartup-notification
thunar	libxfce4ui, exo, libpng, gtk+, glib	libexif, gdbus, libnotify, libstartup-notification, freetype, libjpeg, libxfce4panel, xfconf
xfce4-settings	libxfce4ui, exo, xfconf, gdbus, libXi, libXrandr	libXcursor, libnotify
xfce4-session	libxfce4ui, perl-xml-parser, libwnck, gdbus	
xfwm4	libxfce4util, libxfce4ui, libxfconf, gdbus, libwnck	libstartup-notification
xfdesktop	libxfce4ui, xfconf, libwnck, exo garcon	thunarx, garcon, libnotify
xfce4-appfinder	libxfce4ui, garcon, gio	
tumbler	dbus-glib, gio, gmodule, gdk-pixbuf	freetype-config (fonts), jpeg (jpeg thumbnails from exif), libffmpegthumbnailer (video), gstreamer-1.0 (video), poppler-glib (pdf), libgsf (odf), libopenraw-gnome (various raw images)

[Back To Top](#)

---

## 3.4 Building the packages

### 3.4.1 Installation location and variables

Now it is finally time to start compiling the Xfce desktop. Last remaining thing is to choose an installation location. Possible examples are `/usr`, `/usr/local` and `/opt/xfce4` for a system-wide installation or `$HOME/local` if you want to install in your home directory (whatever you choose, never put spaces in the name). We will refer to this location as `${PREFIX}` in the code examples below!

You need to make sure the `PKG_CONFIG_PATH` variable include the path to the `*.pc` files installed by the Xfce libraries:

```
export PKG_CONFIG_PATH="${PREFIX}/lib/pkgconfig:$PKG_CONFIG_PATH"
```

You can also set some optimization flags for the compiler. This may speed up Xfce, but can also make debugging impossible on some systems. Even worse, this can also cause gcc to generate broken code, so be careful with this. The line below should be safe for most system.

```
export CFLAGS="-O2 -pipe"
```

If you want a debugging environment, unset the `CFLAGS` and use `--enable-debug`, see below for the different debug levels.

```
export CFLAGS=""
```

### 3.4.2 Compiling and installing

Next step is compiling the various packages following the dependency chain above. If you have downloaded the tarballs, you will need to unpack them before going on. Inside each package directory you should run the following command:

```
./configure --prefix=${PREFIX} && make && make install
```

To build from GIT you need to install the `xfce4-dev-tools` package first, all other packages should be installed with this command:

```
./autogen.sh --prefix=${PREFIX} && make && make install
```

Specific configure options for each package can be shown with `./configure --help`. Note that most packages will see a performance benefit if passed the configure option `--disable-debug`.

However, if you want to provide backtraces or test new code, no `$CFLAGS`, no binary stripping and `--enable-debug=full` are recommended. Please note that `--disable-debug` is not available for `xfce4-dev-tools`.

---

**Note:** When reusing the `./configure` script while building from GIT remember to pass the flag `--enable-maintainer-mode`.

---

If you install the package in a public prefix, for example `/usr` or `/usr/local`, you will need to run `make install` using `sudo`. This way, you will be asked for the system administrator password and will have write permissions to install in those locations.

```
sudo make install
```

### 3.4.3 Understanding Debug levels

Most Xfce modules use the m4 macro **XDT\_FEATURE\_DEBUG** to manage the debug compiler levels. If so there are a number of options possible for `--enable-debug=`

#### no

Cast checks and asserts in the Glib macros are disabled, this might be a tad faster, but it can results in segfaults and unexpected crashes.

#### minimum

This is the same as `--disable-debug`. No additional compiler options are added, but checks in Glib are not disabled. This is the recommended level for users and distributions.

#### yes

A number of `$CFLAGS` are set to check the code for different errors. During configure you'll see all the options are tested to avoid binaries that do not work.

#### full

Same as the level above, including `-Werror` so the compilers aborts during compiler errors. The compiler will also generate binaries with debug information suitable for `[[contribute:bugs:|backtraces]]` and memory checks. This is the recommended level for developers or users who want to report bugs.

Back To Top

---

## 3.5 Finding additional Help

Hopefully everything went fine after reading this guide. If you still have problems or questions you can ask on the [Xfce users mailing list](#), the [forum](#) or ask distribution related questions on their forums and mailing lists.

Have fun building Xfce!

Back To Top

---

[start](#)|[Back to main Xfce documentation page](#)

## HOW TO CONTRIBUTE TO XFCE

Xfce is developed by a small group of hard-working volunteers. There are many ways in which you can contribute to Xfce, all of which are most welcome!

For a more in-depth look at how to contribute to Xfce, it is recommended to read [The Ultimate Contributor's Guide to Xfce](#) for additional helpful information.

If you would like to donate to Xfce, please see the [\[\[start#Donations|Donations\]\]](#) section.

### Contents

- *How to contribute to Xfce*
  - *Bug Reporting and Testing*
  - *Communication*
  - *Development*
  - *Documentation*
  - *Financial Contributions*
  - *Feature Requests*
  - *Translation*

---

### 4.1 Bug Reporting and Testing

One of the most useful tasks that we rely on the community for is testing and reporting of bugs.

- You can report bugs on [Xfce GitLab site](#)
- You can ask on our IRC channel [#xfce on Libera.chat](#) for help. A [web irc interface](#) is also available if you don't have an irc client.
- Please do not use the forums to report bugs.

Before submitting a bug, please try your best to check if it has already been reported. When writing your bug report, try to be as descriptive as possible, but avoid verbosity; Mozilla has a nice guide on [how to write a bug report](#).

For more information, see [\[\[:bugs/start|Bug Reporting and Fixing\]\]](#).

- For a list of currently open bugs, see the [\[\[:contribute:bugs:bug-summary|open bugs summary\]\]](#).

If you want to test the current state of Xfce 4.16, you can look at the [Xfce-Test Quick Start Guide](#), which provides instructions for running a Docker container with the latest code from git master as well as the ability to run [Behave](#) (behaviour-driven development) tests.

- As of November 2020, OpenSUSE is now providing a source for “Xfce’s latest builds based on openSUSE’s rolling release Tumbleweed”. This is available as a QEMU based virtual machine.
  - [Xfce Virtual Machine Images For Development](#) provides additional information for its installation and bug reporting instructions.
- 

## 4.2 Communication

There are various ways to get in touch with Xfce developers:

See the [Xfce Development Information](#) page for information on contacting the Xfce development team.

---

## 4.3 Development

See the [\[:contribute:dev:start|Xfce Development Information\]](#) page for details on getting a contributor’s account, working with Git and Xfce source code, and instructions for making releases. As well check the [Xfce Wiki](#) more development hints, tips and tricks.

---

## 4.4 Documentation

The [Xfce Wiki](#) and [Documentation](#) contain many helpful pages that require maintaining and updating with every new release. The wiki is freely available for anyone to edit, refer [\[\[documentation|here\]\]](#) for information on how to contribute to the documentation. The [API reference](#) is automatically updated on a weekly basis based on the latest stable releases.

---


## 4.5 Financial Contributions

As of May 2021, Xfce has moved its fiscal hosting to OpenCollective. We are now able to accept financial contributions in both US Dollars as well as Euros. If you wish to donate to Xfce in USD or with PayPal, you can do so by visiting [Xfce’s OpenCollective page](#). Alternatively, if you wish to contribute in Euros, visit [Xfce’s s EU OpenCollective page](#). Every donation is greatly appreciated, but please remember, this is an open source project that is 100% run by volunteers, no one is implicitly obliged to act upon your requests.

For contributions in USD (SWIFT transfers / PayPal / credit card):

[\[\[https://opencollective.com/xfce/donate|{\[:contribute:oc\\_donate\\_button30px.png\]}\]\]](#) Note: For financial contributions to the Xfce Collective, there is a 10% transaction fee.

For contributions in EUR (SEPA transfers / credit card):

<https://opencollective.com/xfce-eu/donate> Note: For financial contributions to the Xfce (EU) Collective, there is a 6% transaction fee. Additionally, SEPA transfers may take several business days to process and will not appear in the transaction records immediately.

For a list of our supporters and details on individual transactions, see our [Supporters page](#).

---

## 4.6 Feature Requests

Although the philosophy of Xfce is to find the correct balance between features and lightweight, it is still possible to request new features.

If you plan to request a new feature, please first check on our [issue tracker](#) if someone else already filed a request for it. If not, open a new issue and describe your idea in detail. Be warned: You might think your idea is brilliant, but there is a high possibility that there are major downsides.

Obviously, the best way to get your new feature integrated is, to provide a merge request that implements it !

Please note that some components are rather unmaintained, so you might not always get a reply.

---

## 4.7 Translation

Applications, manuals, documentation and many other aspects of Xfce are all translated into different languages. Translations are handled in [Transifex](#), a web-based translation platform. Visit the [translation docs](#) to get started with translating.

[Back To Top](#)

---

[Back to main Xfce documentation page](#)





## BUG REPORTING AND FIXING

One of the most useful tasks that we rely on the community for is the testing and reporting of bugs. Since Xfce runs on various platforms and in a variety of different setups, testing all changes, in every possible situation, is an impossible task. As such, we kindly ask our users to assist us in testing, and reporting all bugs that they may find, using our [issue tracker](#).

- [\[:contribute:bugs:bug-summary|Current Xfce open bugs\]](#)

### Contents

- *Bug Reporting and Fixing*
  - *Crashes*
  - *Search First*
  - *Report*
  - *Backtraces*
  - *Fixing Bugs*

---

## 5.1 Crashes

We always try to make the software as stable as possible, but there is always a possibility a crash or segfault can occur. To problem with crashes is that they are not always reproducible by the developers, this often makes it quite hard to track-down the problem and patch the issue.

Nonetheless, there are a couple of things you can do to help resolve them.

---

## 5.2 Search First

Search the [Open Bug Summary](#) to see if the issue is already known. Also, look through the resolved bugs because the issue might be fixed already, but has not been released or applied downstream. Peeking the [GIT repository](#) might help for the latter.

- For a list of currently open bugs, see the [\[:contribute:bugs:bug-summary|open bugs summary\]](#).
- 

## 5.3 Report

When writing your bug report, try to be as descriptive as possible, but avoid verbosity; Mozilla has a nice guide on [how to write a bug report](#).

If you have not found anything or missed it, report it in the bug tracker. Reports that only say *application X crashed* will be closed as invalid quickly, so try to provide a way to describe and reproduce the problem: *Application X crashed when doing Y*.

In the description provide a short way to reproduce the crash. Also run the application in a terminal and look if there are console warnings, put those in the bug as well.

If the developer cannot reproduce it, he might ask for a *backtrace*, read more about creating backtraces in the next section.

[Back To Top](#)

---

## 5.4 Backtraces

If you do get an Xfce application to crash, it's best to detail exactly what you were doing that led to the crash. In addition, we need the `//backtrace//` of the crash. However most distributions by default make backtrace output useless because their binaries are optimized and do not contain human-readable information to make a relation between the compiled code and the source code (like file, function and variable-names).

The sites below provide guidance from the respective distributions how to remedy this situation so you can provide all the info we need to fix your issue.

In case of panel plugins there is some additional information on [\[:xfce:xfce4-panel:debugging|how to start a panel plugin inside a debugger\]](#).

- [https://wiki.archlinux.org/title/Debugging/Getting\\_traces](https://wiki.archlinux.org/title/Debugging/Getting_traces)
- <https://wiki.debian.org/HowToGetABacktrace>
- <https://wiki.ubuntu.com/DebuggingProgramCrash>
- [http://wiki.gentoo.org/wiki/Project:Quality\\_Assurance/Backtraces](http://wiki.gentoo.org/wiki/Project:Quality_Assurance/Backtraces)
- <http://fedoraproject.org/wiki/StackTraces>
- [https://wiki.mageia.org/en/Debugging\\_software\\_crashes](https://wiki.mageia.org/en/Debugging_software_crashes)
- [http://wiki.mandriva.com/en/Development/Howto/Software\\_Crash](http://wiki.mandriva.com/en/Development/Howto/Software_Crash)
- [http://en.opensuse.org/openSUSE:Bugreport\\_application\\_crashed](http://en.opensuse.org/openSUSE:Bugreport_application_crashed)
- [https://wiki.sabayon.org/?title=Debugging\\_Symbols\\_-\\_splitdebug](https://wiki.sabayon.org/?title=Debugging_Symbols_-_splitdebug)

[Back To Top](#)

---

## 5.5 Fixing Bugs

Once a bug has been found, the cause of the bug needs to be tracked down, and then (obviously) fixed. If you want to get involved in the actual development process of Xfce a great way to start is by solving bugs and attaching a patch file to the reported bug. ( “git format-patch” is the preferred way to create a patch file )

Not familiar with git? [\[\[contribute/dev/git/start|This manual\]\]](#) will help you to get started.

To get started, best read our [\[\[contribute/dev/coding/example|beginners example on how to fix a simple xfce bug\]\]](#).

[Back To Top](#)

---

[start:](#)[Return to Main Xfce documentation page](#)



## CORE MODULES

Application Finder (xfce4-appfinder) – Application to quickly run applications and commands  
Configuration Storage System (xfconf) – D-Bus-based configuration storage system  
Desktop Manager (xfdesktop) – Configure the desktop background image, icons, launchers and folders  
Development Tools (xfce4-dev-tools) – A set of scripts and m4/autoconf macros that ease build system maintenance.  
Helper Applications (exo) – Manage preferred applications and edit .desktop files  
File Manager (thunar) – The fast and easy to use file manager for the Xfce Desktop  
Menu Library (garcon) – Library used for menu implementation  
Panel (xfce4-panel) – Application launchers, window buttons, applications menu, workspace switcher and more  
Power Manager (xfce4-power-manager) – Manage power sources and power consumption of devices  
Session Manager (xfce4-session) – Save the state of your desktop and restore it on the next startup  
Settings Manager (xfce4-settings) – The Settings daemon which persists many Xfce settings  
Thumbnail Service (tumbler) – A D-Bus service for applications to request thumbnails for various URI schemes and MIME types  
Utility Sharing Library (libxfce4util) – Library used to share commonly used non-GTK+ utilities among the Xfce applications  
Widget Sharing Library (libxfce4ui) – Used to share commonly used Xfce widgets among the Xfce applications  
Window Manager (xfwm4) – Handles the placement of windows on the screen

### 6.1 xfce4-appfinder - Application Finder

#### Contents

- *xfce4-appfinder - Application Finder*
  - *General*
  - *Debugging Support*
  - *Latest Release*
  - *Source Code Repository*
  - *Reporting Bugs*

The Application Finder is a program to find and launch installed applications on your system, and quickly execute commands. It does this by searching your file system for .desktop files, and displays a categorized list of all the GUI applications on your system.

**Tip:** You can use the Application Finder to quickly add [[:xfce:xfce4-panel:launcher|launchers]] to your [[:xfce:xfce4-panel:panel]] - simply drag an item onto your desired panel and it will automatically create a launcher for that application.</note>

### 6.1.1 General

- [\[\[usage|Usage\]\]](#) – Overview of the Application Finder
  - [\[\[preferences|Preferences\]\]](#) – Configure the behaviour of the application and create custom actions
  - [\[\[examples|Custom Action Examples\]\]](#) – Examples for custom actions
- 

### 6.1.2 Debugging Support

xfce4-appfinder currently supports three different levels of debugging support, which can be setup using the configure flag `--enable-debug` (check the output of `configure --help`):

Argument	Description
yes	This is the default for Git snapshot builds. It adds all kinds of checks to the code, and is therefore likely to run slower. Use this for development of xfce4-appfinder and locating bugs in xfce4-appfinder.
minimum	This is the default for release builds. <b>This is the recommended behaviour.</b>
no	Disables all sanity checks. Don't use this unless you know exactly what you do.

---

### 6.1.3 Latest Release

{ {rss><https://archive.xfce.org/feeds/project/xfce4-appfinder> 1 date description 2h} }

[Previous Releases](#)

---

### 6.1.4 Source Code Repository

<https://gitlab.xfce.org/xfce/xfce4-appfinder>

---

### 6.1.5 Reporting Bugs

- [\[\[bugs|Reporting Bugs\]\]](#) – Open bug reports and how to report new bugs
- 

[start](#)|[Back to main Xfce documentation page](#)

## 6.2 xfconf - Configuration Storage System

### Contents

- *xfconf - Configuration Storage System*
  - *Introduction*
  - *Accessing Configuration Data*
  - *Kiosk Mode*
  - *Latest Release*
  - *Source Code Repository*
  - *Reporting Bugs*

### 6.2.1 Introduction

Xfconf is a hierarchical (tree-like) configuration system where the immediate child nodes of the root are called **channels**. All settings beneath the channel nodes are called **properties**.

Valid channel and property names are composed of the ASCII US-English uppercase characters A-Z, lowercase characters a-z, numerals 0-9, the dash (-), and underscore (\_). No other characters are allowed for channel names. The less than (<) and greater than (>) characters (aka angle brackets) are also allowed in property names, but not in channel names.

Property names are referenced by their full path underneath their channel, for example: `/main/history-window/last-accessed`. Of course, when querying a particular property, the channel must be specified separately as well.

Both channel and property names are case-insensitive. For example, the following four all refer to the same property:

- Channel: ExampleApp, property: `/main/history-window/last-accessed`
- Channel: EXAMPLEAPP, property: `/main/history-window/last-accessed`
- Channel: ExampleApp, property: `/Main/History-Window/Last-Accessed`
- Channel: exampleapp, property: `/MAIN/history-window/last-accessed`

### 6.2.2 Accessing Configuration Data

Settings stored in Xfconf can be accessed in numerous ways:

- From within applications of Settings Manager. Most of the time, settings stored in Xfconf and configuration options presented in the GUI are tightly coupled. It means that any action of the GUI settings is automatically and immediately propagated to the configuration system and vice-versa. See also: `[[xfce:xfce4-settings:xfsettingsd|Settings Daemon]]`.
- From Command Line Interface (CLI) using `[[[:xfce:xfconf:xfconf-query:|xfconf-query]]`.
- From GUI. Settings, which are not exposed in configuration dialogs, can be browsed and manipulated using `[[xfce:xfce4-settings:editor|Settings Editor]]`.

- If everything else fails, settings can be edited manually. Xfconf stores all its data in `[[xml-file-format|XML files]]`, which can be edited when Xfconf is not running. Be very careful when using this option, as it is easy to make the file unparseable or inconsistent.
- 

### 6.2.3 Kiosk Mode

System administrators can lock certain values or entire channels to prevent users from changing them. In this case, the administrator may provide default values that will be used regardless of what may appear in a user-writable configuration file. If the sysadmin does not wish to provide values, but only wants to lock a channel/property, the application's default fallback value will be used instead.

Locking a channel or property is as simple as creating a configuration file and setting either (and only either) the `locked` or `unlocked` attribute on `<channel>` or `<property>` elements that should be restricted. This configuration file should be placed in a system location that is read by the daemon (see `File locations` above).

Both the “locked” and `unlocked` attributes take a semicolon-separated list of system user and group names. User names should be entered as-is, and group names should be entered with an `@` symbol prepended to the group name.

The “locked” attribute specifies users and groups who may not modify the property. The `unlocked` attribute specifies users and groups who may modify the property, with other users locked out. Only one of the two attributes may be specified for a particular channel or property. If both are present, the `unlocked` attribute is used, and the `locked` attribute is ignored.

Note that `<channel>` locking locks all properties under that channel, but `<property>` locking locks only the property with the `locked` or `unlocked` attribute; none of the sub-properties are locked unless they also contain a `locked` or `unlocked` attribute.

See `[:xfce:xfce4-session:advanced#kiosk_mode|Xfce-Session Kiosk Mode]` for specific parameter used in configuring kiosk mode.

---

### 6.2.4 Latest Release

`{ {rss}>https://archive.xfce.org/feeds/project/xfconf 1 date description 2h }`

[Previous Releases](#)

---

### 6.2.5 Source Code Repository

<https://gitlab.xfce.org/xfce/xfconf>

---



## 6.2.6 Reporting Bugs

- [\[:xfce:xfconf:bugs|Reporting Bugs\]](#) – Open bug reports and how to report new bugs

[\[:xfce:xfce4-settings:start:|Return to main xfce4-settings page\]](#)

[\[:start|Back to main Xfce documentation page\]](#)

## 6.3 xfdesktop - Xfce Desktop Manager

### Contents

- *xfdesktop - Xfce Desktop Manager*
  - *General*
  - *Minimum Requirements*
  - *Debugging Support*
  - *Latest Release*
  - *Source code repository*
  - *Reporting Bugs*

Xfdesktop is a desktop manager for the Xfce Desktop Environment. It handles the following tasks:

- background image / color
- root menu, window list
- minimized app icons
- file icons on the desktop (using Thunar libs)

It can bring up an applications menu and a list of all running applications when you click on the desktop with the right or middle mouse button respectively. Settings are available via the [\[:xfce4-settings:|Settings Manager\]](#).

### 6.3.1 General

- [\[:usage|Usage\]](#) – Overview of the Desktop Manager
- [\[:preferences|Preferences\]](#) – Configuring the desktop
- [\[:command-line|Command-line Options\]](#) – Parameters for using “xfdesktop” in the command-line
- [\[:advanced|Advanced\]](#) – Hidden options

### 6.3.2 Minimum Requirements

- intltool 0.31
  - GTK+ 3.22.0
  - libxfce4util 4.13
  - libxfce4ui 4.13
  - libwnck 3.14
  - libexo 0.11
  - xfconf 4.12.1
  - garcon 0.6.0 (optional; required for apps menu)
  - thunar 1.7.0 (optional; required for file icons)
  - tumbler 1.6 (optional; enables thumbnail previews for file icons)
  - cairo 1.12
- 

### 6.3.3 Debugging Support

xfdesktop currently supports three different levels of debugging support, which can be setup using the configure flag `--enable-debug` (check the output of `configure --help`):

Argument	Description
yes	This is the default for Git snapshot builds. It adds all kinds of checks to the code, and is therefore likely to run slower. Use this for development of xfdesktop and locating bugs in xfdesktop.
minimum	This is the default for release builds. <b>This is the recommended behaviour.</b>
no	Disables all sanity checks. Don't use this unless you know exactly what you do.

---

### 6.3.4 Latest Release

```
{ {rss>https://archive.xfce.org/feeds/project/xfdesktop 1 date description 2h} }
```

[Previous Releases](#)

---

### 6.3.5 Source code repository

<https://gitlab.xfce.org/xfce/xfdesktop>

### 6.3.6 Reporting Bugs

- [\[:xfce:xfdesktop:bugs|Reporting Bugs\]](#) – Open bug reports and how to report new bugs

:start|Back to main Xfce documentation page

## 6.4 xfce4-dev-tools

### Contents

- *xfce4-dev-tools*
  - *xfce-build containerized build environment*
  - *CI templates for Xfce*
  - *Release notes for 4.15.0*
  - *Latest Release*
  - *Source Code Repository*
  - *Reporting Bugs*

The Xfce development tools are a collection of tools and macros for Xfce developers and people that want to build Xfce from Git. In addition, it contains the Xfce developer's handbook.

### 6.4.1 xfce-build containerized build environment

This project also contains the code to build and deploy `xfce-build` to the [xfce-build area on Docker Hub](#). This container is the build environment used by Xfce to build the various projects. It can also be used as your own build environment as follows:

```
docker run --rm -u $(id -u ${USER}):$(id -g ${USER}) \
--volume $(pwd):/tmp xfce/xfce-build:master /bin/bash \
-c "cd /tmp; ./autogen.sh && make distcheck"
```

The steps to build the container are encoded in the `[Dockerfile]` (Dockerfile) in this repository, and is built via the build job in `[.gitlab-ci.yml]` (`gitlab-ci.yml`).

## 6.4.2 CI templates for Xfce

The [CI folder](ci/) contains the `build_project.yml` template for building the various Xfce projects, as well as supporting scripts such as `build_libs.sh` which handles building any needed dependencies. This helps us avoid repeating the same build code in each project.

---

## 6.4.3 Release notes for 4.15.0

[Please note that this is a development release.]

- New macro `XDT_VERSION_INIT(SEMVER[, TAG])`

This macro takes care of setting up the version numbering. It will define the following macros based on SEMVER and TAG:

- `xdt_version`
- `xdt_version_major`
- `xdt_version_minor`
- `xdt_version_micro`
- `xdt_version_tag`
- `xdt_version_build`
- `xdt_debug_default`

If TAG isn't specified, the `xdt_version_tag` and `xdt_version_git` will be empty and `xdt_debug_default` will be set to "minimum", otherwise the `xdt_version_build` will contain a git hash and `xdt_debug_default` will be set to "yes"

Example usage:

```
XDT_VERSION_INIT([4.15.3],[git])
AC_INIT([xfce4-someproject],[xdt_version()])
...
XDT_FEATURE_DEBUG([xdt_debug_default])
```

- `XDT_I18N` macro will now automatically find the linguas if none are specified during the autoconf run.

- Install m4 macros in default search dir. This makes autoconf find the macros by default.

By setting version via `XDT_VERSION_INIT` and using `XDT_I18N` without arguments, this is no longer necessary to use a separate `configure.ac.in` and a plain `configure.ac` can be used ! A big thank to Natanael Copa (ncopa) for his work on this part !

- `xfce4-dev-tools` now also handle the CI (docker container, .gitlab-ci.yml templates files etc) on our new gitlab instance. A merge request or a push to master to any of our projects automatically trigger a build. This will help us to catch errors quicker and have better quality. Thanks to Jason Yavorska and Simon Steinbeiss.

- New helpers are also available (`xfce-get-release-notes`, `xfce-get-translations`, `xfce-update-news`) to help our maintainers to do releases

---

### 6.4.4 Latest Release

{ {rss><https://archive.xfce.org/feeds/project/xfce4-dev-tools> 1 date description} }

[Previous Releases](#)

---

### 6.4.5 Source Code Repository

<https://gitlab.xfce.org/xfce/xfce4-dev-tools>

---

### 6.4.6 Reporting Bugs

- [\[\[:xfce:xfce4-dev-tools:bugs|Reporting Bugs\]\]](#) – Open bug reports and how to report new bugs
- 

[:start|Back to main Xfce documentation page](#)

## 6.5 Exo - Helper Applications

### Contents

- *Exo - Helper Applications*
  - *Applications*
  - *Required Packages*
  - *Debugging Support*
  - *Latest Release*
  - *Source Code Repository*
  - *Reporting Bugs*

---

Exo is an extension library used in the Xfce desktop, originally developed by [os-cillation](#). It has some helper applications that are used throughout the entire Xfce desktop to manage preferred applications and edit `.desktop` files.

exo is targeted at application development and is considered ready for production use. You no longer need to define the `EXO_API_SUBJECT_TO_CHANGE` preprocessor symbol.

---

### 6.5.1 Applications

- `[[desktop-item-edit|Desktop Item Editor (exo-desktop-item-edit)]]` – Editor for .desktop files
  - `[[preferred-applications|Preferred Applications (exo-preferred-applications)]]` – Select the default applications to be used for various services, i.e. the web browser that should be used to open hyperlinks
- 

### 6.5.2 Required Packages

- GIO
  - GLib
  - GTK+
  - gthread
  - libxfce4ui
  - libxfce4util
  - gio-unix
- 

### 6.5.3 Debugging Support

libexo currently supports four different levels of debugging support, which can be setup using the configure flag `--enable-debug` (check the output of `configure --help`):

<code>`full'</code>	This is the strongest option. Deprecation and some coding style checks are made and will cause the build to fail for any errors or warnings.
<code>`yes'</code>	This is the default for Git snapshot builds. It adds all kinds of checks to the code, and is therefore likely to run slower. Use this for development of libexo and locating bugs in libexo.
<code>`minimum'</code>	This is the default for release builds. This is the recommended behavior.
<code>`no'</code>	Disables all sanity checks. Don't use this unless you know exactly what you are doing.

---

## 6.5.4 Latest Release

{ {rss><https://archive.xfce.org/feeds/project/exo> 1 date description 2h} }

[Previous Versions](#)

---

## 6.5.5 Source Code Repository

<https://gitlab.xfce.org/xfce/exo>

---

## 6.5.6 Reporting Bugs

- [\[\[bugs|Reporting Bugs\]\]](#) – Open bug reports and how to report new bugs
- 

[:start|Back to main Xfce documentation page](#)

# 6.6 Thunar File Manager

### Contents

- *Thunar File Manager*
  - *Introduction*
  - *Thunar Plugins*
  - *Debugging Support*
  - *Standards compliance*
  - *Required Packages*
  - *Latest Release*
  - *Source Code Repository*
  - *Reporting Bugs*

## 6.6.1 Introduction

//Thunar// is a modern file manager for the Xfce Desktop Environment. //Thunar// has been designed from the ground up to be fast and easy to use. Its user interface is clean and intuitive and does not include any confusing or useless options by default. //Thunar// starts up quickly and navigating through files and folders is fast and responsive.

- [\[\[the-file-manager-window|The File Manager Window\]\]](#) – working with the Thunar window and modifying the layout

- [\[\[working-with-files-and-folders|Working with Files and Folders\]\]](#) – basic file handling
  - [\[\[using-removable-media|Using Removable Media\]\]](#) – working with removable devices
  - [\[\[preferences|File Management Preferences\]\]](#) – details of the different file manager preferences
  - [\[\[faq|Frequently Asked Questions\]\]](#) – some random things you probably want to know
  - [\[\[hidden-settings|Advanced Settings\]\]](#) – The xfce4-settings-editor, customize gtk3 css and more possible tweaks
  - [\[\[unix-filesystem|The UNIX filesystem\]\]](#) – understanding the UNIX File System
  - [\[\[send-to|The “Send To” Menu\]\]](#) – how to customize the send to menu
- 

## 6.6.2 Thunar Plugins

Thunar plugins can be installed as separate, additional packages in order to extend the functionality of Thunar. Most Thunar plugins offer additional options on files in the context menu or via shortcut. The interface which is used by thunar plugins is the [thunarx API](#).

- [\[\[:bulk-renamer:|Bulk Renamer\]\]](#) – rename multiple files at once
  - [\[\[custom-actions|Custom Actions\]\]](#) – custom commands associated with common mime-types or extensions
  - [\[\[archive|Archive Plugin \(thunar-archive-plugin\)\]\]](#) – allows to create and extract archive files
  - [\[\[media-tags|Media Tags Plugin \(thunar-media-tags-plugin\)\]\]](#) – improved support for ID3 tags
  - [\[\[thunar-shares-plugin|Shares Plugin \(thunar-shares-plugin\)\]\]](#) – Quickly share folders using Samba without requiring root access
  - [\[\[thunar-volman|Volume Manager \(thunar-volman\)\]\]](#) – Automatic management of removable devices in Thunar
  - [\[\[thunar-vcs-plugin|VCS Plugin \(thunar-vcs-plugin\)\]\]](#) – adds Subversion and GIT actions to the context menu
- 

Other Thunar Related Packages

- [\[\[unix-filesystem#gnome\\_virtual\\_file\\_system|Gnome Virtual File System\]\]](#) – required for trash support, mounting removable media, and [\[\[unix-filesystem#remote\\_file\\_systems|remote filesystems\]\]](#)
  - [\[\[:xfce:tumbler:start:|Tumbler\]\]](#) – the remote thumbnail service for Thunar
- 

## 6.6.3 Debugging Support

Thunar currently supports four different levels of debugging support, which can be setup using the configure flag `--enable-debug` (check the output of `configure --help`):



Argument	Description
full	Use this if you want to debug Thunar to locate a bug. The build will most probably be noticeably slower. This is also recommended for people that want to develop Thunar stuff.
yes	Adds all kinds of checks to the code, and is therefore likely to run slower. Use this if you want to develop for Thunar (e.g. writing Thunar extensions and such).
minimum	This is the default for release builds, and presents the recommended behaviour.
no	Disables all sanity checks. Don't use this unless you know exactly what you do.

For some additional build & debug hints, as well check the [Thunar Wiki](#) pages.

---

## 6.6.4 Standards compliance

Thunar supports the following standards/specifications:

- [XDG Base Directory Specification](#)
  - [Shared MIME Database Specification](#)
  - [X Direct Save \(XDS\) Protocol for the X Window System](#)
  - [Icon Theme Specification](#)
  - [Thumbnail Managing Standard](#)
  - [File URI Specification](#)
  - [Desktop Trash Can Specification](#)
- 

## 6.6.5 Required Packages

Thunar depends on the following packages:

- [perl](#)
- [GTK+](#)
- [GLib](#)
- [exo](#)
- [libxfce4util](#)
- [libxfce4ui](#)
- [xfconf](#)
- [intltool](#)

Thunar can optionally be build with the following packages:

- [D-Bus](#) (strongly suggested, required for inter process communication)
- [xfce-panel](#) (required to build the trash panel applet)
- [gudev](#) (required for thunar-volman)

- [libnotify](#) (required for mount notification support)

For concrete information on the minimum required versions, check [configure.ac.in](#)

---

## 6.6.6 Latest Release

```
{ {rss>https://archive.xfce.org/feeds/project/thunar 1 date description 2h} }
```

[Previous Releases](#)

---

## 6.6.7 Source Code Repository

<https://gitlab.xfce.org/xfce/thunar>

For additional build, test and debugging tips, see [\[\[https://wiki.xfce.org/thunar/dev/build\\_and\\_run|this page\]\]](https://wiki.xfce.org/thunar/dev/build_and_run) for more suggestions.

---

## 6.6.8 Reporting Bugs

- [\[\[bugs|Reporting Bugs\]\]](#) – Open bug reports and how to report new bugs
- 

[:start|Back to main Xfce documentation page](#)

## 6.7 garcon - menu library

### Contents

- *garcon - menu library*
  - *Introduction*
  - *Required Packages*
  - *Debugging Support*
  - *Latest Release*
  - *Source Code Repository*
  - *Reporting Bugs*

---

**Garcon** is an implementation of the [freedesktop.org compliant menu](#) specification. It is based on GLib/GIO and aims at covering the entire specification except for legacy menus. It was started as a complete rewrite of the former Xfce menu library called `libxfce4menu`, which, in contrast to `garcon`, was lacking menu merging features essential for loading menus modified with menu editors.

---

### 6.7.1 Introduction

Garcon covers almost every part of the menu specification except for legacy menus and a few XML attributes. In contrast to libxfce4menu, it can also load menus modified with menu editors such as Alacarte as menu merging is now supported. The only crucial feature still missing is monitoring menus and menu items for changes. This is something that will be worked on for the next release.

The garcon API will most likely not be frozen until its 1.0.0 release (It is currently at version 0.7.1)!

---

### 6.7.2 Required Packages

- GIO
  - GLib
  - gobject
  - gthread
  - GTK+
  - libxfce4ui
  - libxfce4util
- 

### 6.7.3 Debugging Support

garcon currently supports three different levels of debugging support, which can be setup using the configure flag `--enable-debug` (check the output of `configure --help`):

---

### 6.7.4 Latest Release

```
{ {rss>https://archive.xfce.org/feeds/project/garcon 1 date description 2h} }
```

[Previous Releases](#)

---

### 6.7.5 Source Code Repository

<https://gitlab.xfce.org/xfce/garcon>

---

## 6.7.6 Reporting Bugs

- [\[:xfce:garcon:bugs|Reporting Bugs\]](#) – Open bug reports and how to report new bugs

---

[:start|Back to main Xfce documentation page](#)

## 6.8 xfce4-panel - Xfce Panel

### Contents

- *xfce4-panel - Xfce Panel*
  - *General*
  - *Internal Plugins*
  - *External Plugins*
  - *Advanced*
  - *Required Packages*
  - *Latest Release*
  - *Source Code Repository*
  - *Reporting Bugs*

---

The Xfce Panel is part of the Xfce Desktop Environment and features application launchers, panel menus, a workspace switcher and more. Many aspects of the panel can be configured through the GUI, but also by GTK+ style properties and hidden Xfconf settings.

---

### 6.8.1 General

- [\[:getting-started|Getting Started\]](#) – Starting the Xfce Panel for the first time
  - [\[:add-new-items|Add New Items\]](#) – Add new plugins to the Xfce Panel
  - [\[:preferences|Preferences\]](#) – Configure the Xfce Panel
  - [\[:https://wiki.xfce.org/howto:customize-menu|Customize the Xfce menu\]](https://wiki.xfce.org/howto:customize-menu) – Customize menu entries
  - [\[:theming|Theming\]](#) – Styling the Xfce Panel
-

## 6.8.2 Internal Plugins

- `[[actions|Action Buttons]]` – Adds system action buttons to the panel
- `[[applicationsmenu|Applications Menu]]` – Adds a menu containing categories of installed applications
- `[[clock|Clock]]` – What time is it? – a clock plugin for the panel.
- `[[directorymenu|Directory Menu]]` – Show a directory tree in a menu
- `[[launcher|Launcher]]` – Program launcher with optional menu
- `[[systray|Notification Area]]` – Area where notification icons appear
- `[[separator|Separator]]` – Adds a separator or space between panel plugins
- `[[showdesktop|Show Desktop]]` – Hide all windows and show the desktop
- `[[tasklist|Window Buttons]]` – Switch between open windows using buttons
- `[[windowmenu|Window Menu]]` – Switch between open windows using a menu
- `[[pager|Workspace Switcher]]` – Switch between virtual desktops

## 6.8.3 External Plugins

- `[:panel-plugins:xfce4-battery-plugin:start|Battery]]` – A battery monitor panel plugin for Xfce4 compatible with APM and ACPI, for Linux and \*BSD.
- `[:panel-plugins:xfce4-calculator-plugin:start|Calculator]]` – a calculator plugin for the Xfce4 panel.
- `[:panel-plugins:xfce4-clipman-plugin:start|Clipman]]` – A clipboard manager for Xfce
- `[:panel-plugins:xfce4-cpufreq-plugin:start|Cpufreq]]` – shows information about the CPU governor and frequencies supported and used by your system.
- `[:panel-plugins:xfce4-cpugraph-plugin:start|Cpugraph]]` – offers multiple display modes (LED, gradient, fire, etc...) to show the current CPU load of the system.
- `[:panel-plugins:xfce4-datetime-plugin:start|Datetime]]` – shows the date and time in the panel, and a calendar appears when you left-click on it.
- `[:panel-plugins:xfce4-diskperf-plugin:start|Diskperf]]` – displays instant disk/partition performance (bytes transferred per second).
- `[:panel-plugins:xfce4-docklike-plugin:start|Docklike]]` – A modern, minimalist taskbar for Xfce.
- `[:panel-plugins:xfce4-embed-plugin:start|Embed]]` – Embed arbitrary application windows into the panel.
- `[:panel-plugins:xfce4-eyes-plugin:start|Eyes]]` – Eyes that spy on you.
- `[:panel-plugins:xfce4-fsguard-plugin:start|Fsguard]]` – checks the chosen mountpoint for free disk space.
- `[:panel-plugins:xfce4-genmon-plugin:start|Genmon]]` – spawns the indicated script/program, captures its output (stdout) and displays the resulting string into the panel.
- `[:panel-plugins:xfce4-indicator-plugin:start|Indicator]]` – small plugin written by Mark Trompell to display information from various applications consistently.
- `[:panel-plugins:xfce4-mailwatch-plugin:start|Mailwatch]]` – multi-threaded, multi-mailbox, multi-protocol plugin for checking mails periodically.
- `[:panel-plugins:xfce4-mount-plugin:start|Mount]]` – a mount/umount utility for the panel.

- `[:panel-plugins:xfce4-mpc-plugin:start|Mpc]` – client plugin for MPD, the Music Player Daemon
  - `[:panel-plugins:xfce4-netload-plugin:start|Netload]` – displays the current load of the network interfaces, currently works on Linux, \*BSD, Sun Solaris, HP\_UX and MacOS X.
  - `[:panel-plugins:xfce4-notes-plugin:start|Notes]` – provides sticky notes for your desktop.
  - `[:panel-plugins:xfce4-places-plugin:start|Places]` – a menu with quick access to folders, documents, and removable media.
  - `[:panel-plugins:xfce4-pulseaudio-plugin:start|PulseAudio Plugin]` – Adjust audio volume and control media players on the Xfce desktop
  - `[:panel-plugins:xfce4-sample-plugin:start|Sample]` – Sample plugin developers can use as a base for new panel-plugins
  - `[:panel-plugins:xfce4-sensors-plugin:start|Sensors]` – a hardware sensors plugin for the panel.
  - `[:panel-plugins:xfce4-smartbookmark-plugin:start|Smartbookmark]` – allows you to do a search directly on Internet on sites like google or debian bugzilla.
  - `[:panel-plugins:xfce4-statusnotifier-plugin:start|Statusnotifier]` – provides a panel area for status notifier items (application indicators).
  - `[:panel-plugins:xfce4-stopwatch-plugin:start|Stopwatch]` – Keep track of elapsed time.
  - `[:panel-plugins:xfce4-systemload-plugin:start|Systemload]` – Displays the current CPU load, the memory in use, the swap space and the system uptime.
  - `[:panel-plugins:xfce4-timer-plugin:start|Timer]` – Lets the user run an alarm at a specified time or at the end of a specified countdown period.
  - `[:panel-plugins:xfce4-time-out-plugin:start|Time-out]` – for taking breaks from the computer every X minutes.
  - `[:panel-plugins:xfce4-verve-plugin:start|Verve]` – A comfortable command line plugin for the Xfce panel.
  - `[:panel-plugins:xfce4-wavelan-plugin:start|Wavelan]` – Display stats from a wireless lan interface (signal state, signal quality, network name (SSID)). It supports NetBSD, OpenBSD, FreeBSD, and Linux.
  - `[:panel-plugins:xfce4-weather-plugin:start|Weather]` – Shows the current temperature and weather condition, using weather data provided by [xoap.weather.com](http://xoap.weather.com).
  - `[:panel-plugins:xfce4-whiskermenu-plugin:start|Whiskermenu]` – A menu that provides access to favorites, recently used, and searching installed applications.
  - `[:panel-plugins:xfce4-xkb-plugin:start|Xkb]` – Setup and use multiple (currently up to 4 due to X11 protocol limitation) keyboard layouts.
- 

### 6.8.4 Advanced

- `[:debugging|Debugging]` – Debugging plugins and the panel
-

### 6.8.5 Required Packages

- `exo`
- `garcon`
- `GIO`
- `GLib`
- `GTK+`
- `libxfce4ui`
- `libxfce4util`
- `libwnck`
- `xfconf`

For specific information on the minimum required versions, check [configure.ac.in](https://configure.ac.in)

---

### 6.8.6 Latest Release

{ {rss><https://archive.xfce.org/feeds/project/xfce4-panel> 1 date description 2h} }

[Previous Releases](#)

---

### 6.8.7 Source Code Repository

<https://gitlab.xfce.org/xfce/xfce4-panel>

---

### 6.8.8 Reporting Bugs

- [\[\[:xfce:xfce4-panel:bugs|Reporting Bugs\]\]](#) – Open bug reports and how to report new bugs
- 

[:start|Back to main Xfce documentation page](#)

## 6.9 xfce4-power-manager - Power Manager

#### Contents

- *xfce4-power-manager - Power Manager*
  - *Introduction*
  - *Additional Information*

- *Required packages*
  - *Latest Release*
  - *Source Code Repository*
  - *Reporting Bugs*
- 

### 6.9.1 Introduction

Xfce4 Power Manager manages the power sources of the computer and connected devices (wireless mice, keyboards, media players etc.). It also allows users to control the display backlight brightness and set power-saving modes for displays and monitors. In addition, xfce4-power-manager provides a set of freedesktop-compliant DBus interfaces to inform other applications about the current power level so that they can adjust their power consumption, and it provides the inhibit interface which allows applications to prevent automatic sleep actions via the power manager; as an example, the operating system's package manager should make use of this interface while it is performing update operations.

---

### 6.9.2 Additional Information

- **[[:xfce:xfce4-power-manager:getting-started|Getting Started]]** – Starting the power manager and some command-line options.
  - **[[:xfce:xfce4-power-manager:preferences|Preferences]]** – Customizing the settings of the power manager.
  - **[[:xfce:xfce4-power-manager:faq|Frequently Asked Questions]]** – Some random things you probably want to know.
  - **[[:xfce:xfce4-power-manager:panel-plugin|Power Manager Panel Plugin]]** – Display your battery's load status, connected devices and control your display backlight brightness.
- 

### 6.9.3 Required packages

Xfce Power Manager depends on the following packages:

- Gtk 3.14 or above.
- Glib 2.42 or above.
- DBus 1.1 or above.
- libxfce4ui 4.12.0 or above.
- libxfce4util 4.12.0 or above
- libnotify 0.4.1 or above.
- xfconf 4.12.0 or above.
- libxfce4panel 4.12.0 or above (optional, for the Xfce panel plugin).
- XRandR 1.2.0 or above.
- DPMS X11 extension (DPMS support, required).



- UPower 0.99.0 or above.
  - Polkit 0.91 or above (optional but recommended).
  - Consolekit (optional but recommended).
  - LoginD/SystemD (optional).
  - Pm-utils (for suspend & hibernate on Linux).
- 

## 6.9.4 Latest Release

{ {rss><https://archive.xfce.org/feeds/project/xfce4-power-manager> 1 date description 2h} }

[Previous Releases](#)

---

## 6.9.5 Source Code Repository

<https://gitlab.xfce.org/xfce/xfce4-power-manager>

---

## 6.9.6 Reporting Bugs

- [\[\[bugs|Reporting Bugs\]\]](#) – Open bug reports and how to report new bugs
- 

[start|Back to main Xfce documentation page](#)

## 6.10 xfce4-session - Session Manager

### Contents

- *xfce4-session - Session Manager*
    - *Introduction*
    - *General*
    - *Latest Release*
    - *Source Code Repository*
    - *Reporting Bugs*
-

## 6.10.1 Introduction

//Xfce4-session// is a session manager for Xfce. Its task is to save the state of your desktop (opened applications and their location) and restore it during a next startup. You can create several different sessions and choose one of them on startup.

---

## 6.10.2 General

- [\[\[preferences|Preferences\]\]](#) – Configure the behaviour of the session manager
  - [\[\[logout|Logout Dialog\]\]](#) – The logout dialog and more information about power management handling
  - [\[\[faq|Frequently Asked Questions\]\]](#) – Tips and tricks related to xfce4-session
  - [\[\[advanced|Advanced\]\]](#) – Advanced information about environment variables and kiosk mode
  - [\[\[knownProblems|Known Issues\]\]](#) – Read this if xfce4-session causes trouble
- 

## 6.10.3 Latest Release

{ {rss><https://archive.xfce.org/feeds/project/xfce4-session> 1 date description 2h} }

[Previous Releases](#)

---

## 6.10.4 Source Code Repository

<https://gitlab.xfce.org/xfce/xfce4-session>

---

## 6.10.5 Reporting Bugs

- [\[\[:xfce:xfce4-session:bugs|Reporting Bugs\]\]](#) – Open bug reports and how to report new bugs
- 

[:start|Back to main Xfce documentation page](#)

## 6.11 xfce4-settings - Settings Manager

### Contents

- *xfce4-settings - Settings Manager*
  - *Settings*
  - *Dialogs*

- [Other Dialogs](#)
- [Latest Release](#)
- [Source Code Repository](#)
- [Reporting Bugs](#)

The `xfce4-settings` component provides a daemon, manager, and editor to centralize the configuration management of the Xfce system.

### 6.11.1 Settings

- [\[\[xfsettingsd|Settings Daemon\]\]](#) – A daemon that runs in the background to apply all the settings
- [\[\[manager|Settings Manager\]\]](#) – Dialog that lists all the settings dialogs provided by this package (listed below) or other core modules of Xfce

### 6.11.2 Dialogs

- [\[\[accessibility|Accessibility\]\]](#) – Improve keyboard and mouse accessibility
- [\[\[appearance|Appearance\]\]](#) – Customize the look of your desktop
- [\[\[color|Color\]\]](#) – Manage the color profiles of your devices
- [\[\[display|Display\]\]](#) – Configure screen settings and layout
- [\[\[keyboard|Keyboard\]\]](#) – Edit keyboard settings and application shortcuts
- [\[\[mouse|Mouse and Touchpad\]\]](#) – Configure pointer device behavior and appearance
- [\[\[editor|Settings Editor\]\]](#) – Editor that allows to modify all the various setting channels
- [\[\[mime|MIME Type Editor\]\]](#) – Editor to modify application associations with MIME types

### 6.11.3 Other Dialogs

The following items are often visible in the settings manager, but not part of the `//xfce4-settings//` package. You can click the links below to redirect to the correct place on this website.

- [\[:xfce:xfce4-panel:preferences|Panel\]\]](#) – Configure the Xfce Panel
- [\[:xfce:xfwm4:workspaces|Workspaces\]\]](#) – Set number and names of workspaces
- [\[:xfce:xfwm4:preferences|Window Manager\]\]](#) – Handles the placement of windows on the screen
- [\[:xfce:xfwm4:wmtweaks|Window Manager Tweaks\]\]](#) – Specified tuning of the window manager
- [\[:xfce:xfce4-session:preferences|Session\]\]](#) – Applications that are started on login
- [\[:xfce:xfdesktop:preferences|Desktop\]\]](#) – Desktop icons and background
- [\[:apps:xfce4-terminal:preferences|Xfce Terminal\]\]](#) – Configure Xfce Terminal

## 6.11.4 Latest Release

{ {rss><https://archive.xfce.org/feeds/project/xfce4-settings> 1 date description 2h} }

[Previous Releases](#)

---

## 6.11.5 Source Code Repository

<https://gitlab.xfce.org/xfce/xfce4-settings>

---

## 6.11.6 Reporting Bugs

- [\[\[xfce:xfce4-settings:bugs|Reporting Bugs\]\]](#) – Open bug reports and how to report new bugs
- 

[:start|Back to main Xfce documentation page](#)

## 6.12 Tumbler - Thumbnail Service

### Contents

- *Tumbler - Thumbnail Service*
  - *Introduction*
  - *Configuration*
  - *Available Plugins*
  - *Standards compliance*
  - *Debugging Support*
  - *Required packages*
  - *Latest Release*
  - *Source Code Repository*
  - *Reporting Bugs*

### 6.12.1 Introduction

Tumbler is a D-Bus service for applications to request thumbnails for various URI schemes and MIME types. It is an implementation of the thumbnail management D-Bus specification described on

<https://wiki.gnome.org/DraftSpecs/ThumbnailerSpec>

written in an object-oriented fashion using GLib and GObject. Its modular architecture makes it very flexible and useful in many situations. It provides plugin interfaces for extending the URI schemes and MIME types for which thumbnails can be generated as well as for replacing the storage backend that is used to store the thumbnails on disk. Tumbler's functionality can also be extended via specialized thumbnailer services implemented in accordance to the thumbnail management D-Bus specification.

Tumbler is used by Thunar, Ristretto, Xfce.

---

### 6.12.2 Configuration

You can override the default cache directory `$HOME/.cache` in which, besides other non-essential files, thumbnails are stored:

- **D-Bus/ systemd**
  - create a file below `“~/config/environment.d/”` and inside set `XDG_CACHE_HOME`. E.g:
  - `“XDG_CACHE_HOME=$HOME/.my_new_cache”`
- **Other**
  - Extend `$HOME/.profile`, `$HOME/.bash_rc` or similar
  - `“export XDG_CACHE_HOME=$HOME/.my_new_cache”`

After that, re-login, make sure the variable is set, and check if it works fine.

Tumbler has a configuration file `tumbler.rc` described in a [\[\[available\\_plugins#configuration|dedicated page\]\]](#).

---

### 6.12.3 Available Plugins

Tumbler allows to load a number [\[\[:xfce:tumbler:available\\_plugins|additional tumbler plugins\]\]](#) to support different file formats. Most plugins are loaded by default, if the required libraries are provided.

---

### 6.12.4 Standards compliance

Tumbler supports and implements the following standards/specifications:

- [XDG Base Directory Specification](#)
  - [Thumbnail Managing Standard](#)
  - [Thumbnail Management D-Bus Specification](#)
  - [File URI Specification](#)
-

## 6.12.5 Debugging Support

Tumbler currently supports four different levels of debugging support, which can be setup using the configure flag `--enable-debug` (check the output of `configure --help`):

``full'` Use this if you want to debug Tumbler to locate a bug. The build will most probably be noticeably slower. This is also recommended for people that want to develop Tumbler stuff.

``yes'` This is the default for git snapshot builds. It adds all kinds of checks to the code, and is therefore likely to run slower. Use this if you want to develop for Tumbler (e.g. writing Tumbler extensions and such).

``minimum'` This is the default for release builds, and presents the recommended behaviour.

``no'` Disables all sanity checks. Don't use this unless you know exactly what you do.

---

## 6.12.6 Required packages

Tumbler depends on the following packages:

- GLib `>= 2.50.0`

Tumbler can optionally use the following packages:

- gdk-pixbuf `>= 2.14.0` (various image formats, some of them requiring the installation of optional libraries specific to gdk-pixbuf; also required for all other plugins below)
  - FreeType `>= 2.0.0` (font)
  - FFmpegthumbnailer `>= 2.0.0` (video)
  - GStreamer `>= 1.0.0` (video)
  - Poppler `>= 0.12` (PDF/PS)
  - libgsf `>= 1.14.9` (ODF)
  - libopenraw `>= 0.0.4` (RAW images)
  - libpng `>= 1.2.0` (for the Thumbnail Managing Standard storage backend)
  - libgepub `>= 0.6.0` (EPUB)
-

## 6.12.7 Latest Release

{ {rss><https://archive.xfce.org/feeds/project/tumbler> 1 date description 2h} }

[Previous Releases](#)

---

## 6.12.8 Source Code Repository

<https://gitlab.xfce.org/xfce/tumbler>

---

## 6.12.9 Reporting Bugs

- [\[\[xfce:tumbler:bugs|Reporting Bugs\]\]](#) – Open bug reports and how to report new bugs
- 

:start: [Return to Main Xfce documentation page](#)

# 6.13 libxfce4util - Utility sharing library

### Contents

- *libxfce4util - Utility sharing library*
    - *Latest Release*
    - *Source Code Repository*
    - *Reporting Bugs*
- 

Libxfce4util is used to share commonly used non-GTK+ utilities among the Xfce applications.

---

## 6.13.1 Latest Release

{ {rss><https://archive.xfce.org/feeds/project/libxfce4util> 1 date description 2h} }

[Releases](#)

---

## 6.13.2 Source Code Repository

<https://gitlab.xfce.org/xfce/libxfce4util>

---

## 6.13.3 Reporting Bugs

- [\[\[:xfce:libxfce4util:bugs|Reporting Bugs\]\]](#) – Open bug reports and how to report new bugs
- 

[:start|Back to main Xfce documentation page](#)

## 6.14 libxfce4ui - widget sharing library

### Contents

- *libxfce4ui - widget sharing library*
  - *Required packages*
  - *Debugging Support*
  - *Latest Release*
  - *Source Code Repository*
  - *Reporting Bugs*

---

The libxfce4ui library is used to share commonly used Xfce widgets among the Xfce applications.

---

### 6.14.1 Required packages

Libxfce4ui depends on the following packages:

- GTK+ 3.18.0 or above
  - GLib 2.42.0 or above
  - Libxfce4util 4.12.0 or above
-



## 6.14.2 Debugging Support

Libxfce4ui currently supports four different levels of debugging support, which can be setup using the configure flag `--enable-debug` (check the output of `configure --help`):

<code>`full'</code>	Use this if you want to debug libxfce4ui to locate a bug. The build will most probably be noticeably slower. This is also recommended for people that want to develop Xfce applications.
<code>`yes'</code>	This is the default for SVN snapshot builds. It adds all kinds of checks to the code, and is therefore likely to run slower. Use this if you want to develop for Xfce.
<code>`minimum'</code>	This is the default for release builds, and presents the recommended behaviour.
<code>`no'</code>	Disables all sanity checks. Don't use this unless you know exactly what you do.

## 6.14.3 Latest Release

```
{ {rss>https://archive.xfce.org/feeds/project/libxfce4ui 1 date description 2h}}
```

Releases

## 6.14.4 Source Code Repository

<https://gitlab.xfce.org/xfce/libxfce4ui>

## 6.14.5 Reporting Bugs

- [\[:xfce:libxfce4ui:bugs|Reporting Bugs\]](#) – Open bug reports and how to report new bugs

[:start|Back to main Xfce documentation page](#)

## 6.15 xfwm4 - Window Manager

### Contents

- *xfwm4 - Window Manager*
  - *General*

- *Configuration*
- *Latest Release*
- *Source Code Repository*
- *Reporting Bugs*

---

xfwm is the window manager for Xfce

---

### 6.15.1 General

- [\[\[introduction|Introduction\]\]](#) – General information about `xfwm4`
  - [\[\[getting-started|Getting Started\]\]](#) – Handling windows and workspaces with `xfwm4`
  - [\[\[faq|Frequently Asked Question\]\]](#) – Tips and tricks related to `xfwm4`
  - [\[\[standards|Supported Standards\]\]](#) – List of freedesktop, Gnome, & KDE specifications supported by `xfwm4`
  - [\[\[https://wiki.xfce.org/howto/install\\_new\\_themes|Installing themes\]\]](https://wiki.xfce.org/howto/install_new_themes) – How to install new themes
  - [\[\[https://wiki.xfce.org/howto/xfwm4\\_theme|Styling themes\]\]](https://wiki.xfce.org/howto/xfwm4_theme) – Styling the Xfce Window Manager
  - [\[\[https://wiki.xfce.org/howto/gtk\\_theme|Xfce GTK Engine Theme How-To\]\]](https://wiki.xfce.org/howto/gtk_theme)
- 

### 6.15.2 Configuration

- [\[\[preferences|Preferences\]\]](#) – Configure window behavior and shortcuts
  - [\[\[wmtweaks|Window Manager Tweaks\]\]](#) – Fine-tune window behavior and effects
  - [\[\[workspaces|Workspaces\]\]](#) – Set number and names of workspaces
- 

### 6.15.3 Latest Release

{ {rss><https://archive.xfce.org/feeds/project/xfwm4> 1 date description 2h} }

[Previous Releases](#)

---

### 6.15.4 Source Code Repository

<https://gitlab.xfce.org/xfce/xfwm4>

---

### 6.15.5 Reporting Bugs

- [\[:xfce:xfwm4:bugs|Reporting Bugs\]](#) – Open bug reports and how to report new bugs

---

[:start|Back to main Xfce documentation page](#)



**APPLICATIONS**



## GETTING STARTED

- [Getting Started with Xfce](#) – Starting point for anyone who wants to use Xfce
- Frequently Asked Questions – General questions relating to Xfce and their answers
- Building Xfce – Instructions on how to compile Xfce from source
- How to Contribute – Detailed information for new contributors
- Bug Reporting and Fixing





## DOCUMENTATION FOR XFCE APPLICATIONS

### 9.1 Core Modules

- Application Finder (xfce4-appfinder) – Application to quickly run applications and commands
- Configuration Storage System (xfconf) – D-Bus-based configuration storage system
- Desktop Manager (xfdesktop) – Configure the desktop background image, icons, launchers and folders
- Development Tools (xfce4-dev-tools) – A set of scripts and m4/autoconf macros that ease build system maintenance.
- Helper Applications (exo) – Manage preferred applications and edit .desktop files
- File Manager (thunar) – The fast and easy to use file manager for the Xfce Desktop
- Menu Library (garcon) – Library used for menu implementation
- Panel (xfce4-panel) – Application launchers, window buttons, applications menu, workspace switcher and more
- Power Manager (xfce4-power-manager) – Manage power sources and power consumption of devices
- Session Manager (xfce4-session) – Save the state of your desktop and restore it on the next startup
- Settings Manager (xfce4-settings) – The Settings daemon which persists many Xfce settings
- Thumbnail Service (tumbler) – A D-Bus service for applications to request thumbnails for various URI schemes and MIME types
- Utility Sharing Library (libxfce4util) – Library used to share commonly used non-GTK+ utilities among the Xfce applications
- Widget Sharing Library (libxfce4ui) – Used to share commonly used Xfce widgets among the Xfce applications
- Window Manager (xfwm4) – Handles the placement of windows on the screen

### 9.2 Applications

- CD Burning application (Xfburn) – A simple CD/DVD burning tool.
- Dictionary (xfce4-dict) – allows you to search different kinds of dictionary services for words or phrases and shows you the results
- File Search Utility (catfish) – a versatile file search utility for the Xfce desktop
- Image Viewer (ristretto) – Image viewer for the Xfce desktop
- Media player (parole) – Media player for the Xfce desktop

- Music Player Daemon (xfmpc) – A graphical GTK+ Music Player Daemon (MPD) client focusing on low footprint
- Notification service (xfce4-notifyd) – Notification service for the Xfce desktop
- Remote filesystem Utility (gigolo) – A frontend to easily manage connections to remote filesystems.
- Panel Profiles (xfce4-panel-profiles) – A simple application to manage Xfce panel layouts
- Screensaver Utility (xfce4-screensaver) – A simple, secure screen saver and locker.
- Screenshot application (xfce4-screenshooter) – Take screenshots of your Xfce desktop
- Taskmanager(xfce4-taskmanager) – Easy to use task manager
- Terminal Emulator (xfce4-terminal) – Terminal emulator for the Xfce desktop
- Text Editor (mousepad) – A simple text editor for Xfce
- Volume Control & Mixer (xfce4-mixer) – A volume control application based on GStreamer & panel-plugin
- Xfdashboard – Provides a Gnome shell-like dashboard for Xfce

Take a look at the list of recommended applications for more light-weight applications, recommended by Xfce users.  
Check the sitemap of this wiki to see all available wiki pages.