

GLFW



Introduction

GLFW is an Open Source, multi-platform library for OpenGL, OpenGL ES and Vulkan application development. It provides a simple, platform-independent API for creating windows, contexts and surfaces, reading input, handling events, etc.

GLFW natively supports Windows, macOS and Linux and other Unix-like systems. An experimental implementation for the Wayland protocol is available but not yet officially supported.

GLFW is licensed under the [zlib/libpng license](#).

The latest stable release is version 3.2.1.

See the [downloads](#) page for details and files, or fetch the `latest` branch, which always points to the latest stable release. Each release starting with 3.0 also has a corresponding [annotated tag](#) with source and binary archives. The [version history](#) lists all user-visible changes for every release.

This is a development branch for version 3.3, which is *not yet described*. Pre-release documentation is available [here](#).

The `master` branch is the stable integration branch and *should* always compile and run on all supported platforms, although details of newly added features may change until they have been included in a release. New features and many bug fixes live in [other branches](#) until they are stable enough to merge.

If you are new to GLFW, you may find the [tutorial](#) for GLFW 3 useful. If you have used GLFW 2 in the past, there is a [transition guide](#) for moving to the GLFW 3 API.

Compiling GLFW

GLFW itself requires only the headers and libraries for your window system. It does not need the headers for any context creation API (WGL, GLX, EGL, NSGL, OSMesa) or rendering API (OpenGL, OpenGL ES, Vulkan) to enable support for them.

GLFW supports compilation on Windows with Visual C++ 2010 and later, MinGW and MinGW-w64, on macOS with Clang and on Linux and other Unix-like systems with GCC and Clang. It will likely compile in other environments as well, but this is not regularly tested.

There are [pre-compiled Windows binaries](#) available for all supported compilers.

See the [compilation guide](#) for more information about how to compile GLFW yourself.

Using GLFW

See the [documentation](#) for tutorials, guides and the API reference.

Contributing to GLFW

See the [contribution guide](#) for more information.

System requirements

GLFW supports Windows XP and later and macOS 10.8 and later. Linux and other Unix-like systems running the X Window System are supported even without a desktop environment or modern extensions, although some features require a running window or clipboard manager. The OSMesa backend requires Mesa 6.3.

See the [compatibility guide](#) in the documentation for more information.

Dependencies

GLFW itself depends only on the headers and libraries for your window system.

The (experimental) Wayland backend also depends on the `extra-cmake-modules` package, which is used to generate Wayland protocol headers.

The examples and test programs depend on a number of tiny libraries. These are located in the `deps/` directory.

- [getopt_port](#) for examples with command-line options
- [TinyCThread](#) for threaded examples
- An OpenGL 3.2 core loader generated by [glad](#) for examples using modern OpenGL
- [linmath.h](#) for linear algebra in examples
- [Nuklear](#) for test and example UI
- [stb_image_write](#) for writing images to disk
- [Vulkan headers](#) for Vulkan tests

The Vulkan example additionally requires the LunarG Vulkan SDK to be installed, or it will not be included in the build. On macOS you need to provide the path to the SDK manually as it has no standard installation location.

The documentation is generated with [Doxygen](#) if CMake can find that tool.

Reporting bugs

Bugs are reported to our [issue tracker](#). Please check the [contribution guide](#) for information on what to include when reporting a bug.

Changelog

- Added `glfwGetError` function for querying the last error code and its description (#970)
- Added `glfwUpdateGamepadMappings` function for importing gamepad mappings in SDL_GameControllerDB format (#900)
- Added `glfwJoystickIsGamepad` function for querying whether a joystick has a gamepad mapping (#900)
- Added `glfwGetJoystickGUID` function for querying the SDL compatible GUID of a joystick (#900)
- Added `glfwGetGamepadName` function for querying the name provided by the gamepad mapping (#900)
- Added `glfwGetGamepadState` function, `GLFW_GAMEPAD_*` and `GLFWgamepadstate` for retrieving gamepad input state (#900)
- Added `glfwGetWindowContentScale`, `glfwGetMonitorContentScale` and `glfwSetWindowContentScaleCallback` for DPI-aware rendering (#235,#439,#677,#845,#898)
- Added `glfwRequestWindowAttention` function for requesting attention from the user (#732,#988)
- Added `glfwGetMonitorWorkarea` function for retrieving the monitor work area (#920,#989,#1322)
- Added `glfwGetKeyScancode` function that allows retrieving platform dependent scancodes for keys (#830)

- Added `glfwSetWindowMaximizeCallback` and `GLFWwindowmaximizefun` for receiving window maximization events (#778)
- Added `glfwSetWindowAttrib` function for changing window attributes (#537)
- Added `glfwGetJoystickHats` function for querying joystick hats (#889,#906,#934)
- Added `glfwInitHint` for setting initialization hints
- Added `glfwWindowHintString` for setting string type window hints (#893,#1139)
- Added `glfwGetWindowOpacity` and `glfwSetWindowOpacity` for controlling whole window transparency (#1089)
- Added `glfwSetMonitorUserPointer` and `glfwGetMonitorUserPointer` for per-monitor user pointers
- Added `glfwSetJoystickUserPointer` and `glfwGetJoystickUserPointer` for per-joystick user pointers
- Added `glfwGetX11SelectionString` and `glfwSetX11SelectionString` functions for accessing X11 primary selection (#894,#1056)
- Added `glfwRawMouseMotionSupported` function for querying raw motion support (#125,#1400,#1401)
- Added headless [OSMesa](#) backend (#850)
- Added definition of `GLAPIENTRY` to public header
- Added `GLFW_TRANSPARENT_FRAMEBUFFER` window hint and attribute for controlling per-pixel framebuffer transparency (#197,#663,#715,#723,#1078)
- Added `GLFW_HOVERED` window attribute for polling cursor hover state (#1166)
- Added `GLFW_CENTER_CURSOR` window hint for controlling cursor centering (#749,#842)
- Added `GLFW_FOCUS_ON_SHOW` window hint and attribute to control input focus on calling show window (#1189)
- Added `GLFW_SCALE_TO_MONITOR` window hint for automatic window resizing (#676,#1115)
- Added `GLFW_JOYSTICK_HAT_BUTTONS` init hint (#889)
- Added `GLFW_LOCK_KEY_MODS` input mode and `GLFW_MOD_*_LOCK` mod bits (#946)
- Added `GLFW_RAW_MOUSE_MOTION` input mode for selecting raw motion input (#125,#1400,#1401)
- Added macOS specific `GLFW_COCOA_RETINA_FRAMEBUFFER` window hint
- Added macOS specific `GLFW_COCOA_FRAME_NAME` window hint (#195)
- Added macOS specific `GLFW_COCOA_GRAPHICS_SWITCHING` window hint (#377,#935)
- Added macOS specific `GLFW_COCOA_CHDIR_RESOURCES` init hint
- Added macOS specific `GLFW_COCOA_MENUBAR` init hint
- Added X11 specific `GLFW_X11_CLASS_NAME` and `GLFW_X11_INSTANCE_NAME` window hints (#893,#1139)
- Added `GLFW_INCLUDE_ES32` for including the OpenGL ES 3.2 header
- Added `GLFW_OSMESA_CONTEXT_API` for creating OpenGL contexts with [OSMesa](#) (#281)
- Added `GenerateMappings.cmake` script for updating gamepad mappings
- Made `glfwCreateWindowSurface` emit an error when the window has a context (#1194,#1205)
- Deprecated window parameter of clipboard string functions
- Deprecated charmods callback
- Removed `GLFW_USE_RETINA` compile-time option
- Removed `GLFW_USE_CHDIR` compile-time option
- Removed `GLFW_USE_MENUBAR` compile-time option
- Removed requirement of at least one window for `glfwWaitEvents` and `glfwPostEmptyEvent` (#1317)
- Bugfix: Calling `glfwMaximizeWindow` on a full screen window was not ignored
- Bugfix: `GLFW_INCLUDE_VULKAN` could not be combined with the corresponding

OpenGL and OpenGL ES header macros

- Bugfix: `glfwGetInstanceProcAddress` returned `NULL` for

``vkGetInstanceProcAddress`` when ``_GLFW_VULKAN_STATIC`` was enabled

- Bugfix: Invalid library paths were used in test and example CMake files (#930)
- Bugfix: The scancode for synthetic key release events was always zero
- Bugfix: The generated Doxyfile did not handle paths with spaces (#1081)
- Bugfix: The gamma ramp generated by `glfwSetGamma` did not use the monitor

ramp size (#1387,#1388)

- [Win32] Added system error strings to relevant GLFW error descriptions (#733)
- [Win32] Removed XInput circular deadzone from joystick axis data (#1045)
- [Win32] Bugfix: Undecorated windows could not be iconified by the user (#861)
- [Win32] Bugfix: Deadzone logic could underflow with some controllers (#910)
- [Win32] Bugfix: Bitness test in `FindVulkan.cmake` was VS specific (#928)
- [Win32] Bugfix: `glfwVulkanSupported` emitted an error on systems with

a loader but no ICD (#916)

- [Win32] Bugfix: Non-iconified full sreeen windows did not prevent screen

blanking or password enabled screensavers (#851)

- [Win32] Bugfix: Mouse capture logic lost secondary release messages (#954)
- [Win32] Bugfix: The 32-bit Vulkan loader library static was not searched for
- [Win32] Bugfix: Vulkan libraries have a new path as of SDK 1.0.42.0 (#956)
- [Win32] Bugfix: Monitors with no display devices were not enumerated (#960)
- [Win32] Bugfix: Monitor events were not emitted (#784)
- [Win32] Bugfix: The Cygwin DLL was installed to the wrong directory (#1035)
- [Win32] Bugfix: Normalization of axis data via XInput was incorrect (#1045)
- [Win32] Bugfix: `glfw3native.h` would undefine a foreign `APIENTRY` (#1062)
- [Win32] Bugfix: Disabled cursor mode prevented use of caption buttons

(#650,#1071)

- [Win32] Bugfix: Returned key names did not match other platforms (#943)
- [Win32] Bugfix: Undecorated windows did not maximize to workarea (#899)
- [Win32] Bugfix: Window was resized twice when entering full screen (#1085)
- [Win32] Bugfix: The HID device notification was not unregistered (#1170)
- [Win32] Bugfix: `glfwCreateWindow` activated window even with `GLFW_FOCUSED`

hint set to false (#1179,#1180)

- [Win32] Bugfix: The keypad equals key was reported as `GLFW_KEY_UNKNOWN`

(#1315,#1316)

- [Win32] Bugfix: A title bar would be drawn over undecorated windows in some

circumstances (#1383)

- [Win32] Bugfix: Standard cursors were not per-monitor DPI aware (#1431)
- [X11] Replaced `_GLFW_HAS_XF86VM` compile-time option with dynamic loading
- [X11] Bugfix: `glfwGetVideoMode` would segfault on Cygwin/X
- [X11] Bugfix: Dynamic X11 library loading did not use full sonames (#941)
- [X11] Bugfix: Window creation on 64-bit would read past top of stack (#951)
- [X11] Bugfix: XDND support had multiple non-conformance issues (#968)
- [X11] Bugfix: The RandR monitor path was disabled despite working RandR (#972)
- [X11] Bugfix: IM-duplicated key events would leak at low polling rates (#747)
- [X11] Bugfix: Gamma ramp setting via RandR did not validate ramp size
- [X11] Bugfix: Key name string encoding depended on current locale (#981,#983)
- [X11] Bugfix: Incremental reading of selections was not supported (#275)
- [X11] Bugfix: Selection I/O reported but did not support `COMPOUND_TEXT`
- [X11] Bugfix: Latin-1 text read from selections was not converted to UTF-8
- [X11] Bugfix: NVidia EGL would segfault if unloaded before closing the display
- [X11] Bugfix: Checking window maximized attrib could crash some WMs (#1356)
- [X11] Bugfix: Update cursor position on enter event (#1366)
- [X11] Bugfix: `glfwSetWindowMonitor` did not update hints when resizing

`non-user-resizable windows`

- [X11] Bugfix: `glfwSetWindowMonitor` did not flush output buffer in some cases
- [X11] Bugfix: `glfwSetWindowMonitor` did not update the EWMH state of hidden

`windows (#1358)`

- [Linux] Added workaround for missing `SYN_DROPPED` in pre-2.6.39 kernel

`headers (#1196)`

- [Linux] Moved to evdev for joystick input (#906,#1005)
- [Linux] Bugfix: Event processing did not detect joystick disconnection (#932)
- [Linux] Bugfix: The joystick device path could be truncated (#1025)
- [Linux] Bugfix: `glfwInit` would fail if inotify creation failed (#833)
- [Linux] Bugfix: `strdup` was used without any required feature macro (#1055)
- [Cocoa] Added support for Vulkan window surface creation via

`[MoltenVK] (https://moltengl.com/moltenvk/) (#870)`

- [Cocoa] Added support for loading a `MainMenu.nib` when available
- [Cocoa] Disabled automatic window tabbing for created windows (#1250)
- [Cocoa] Bugfix: Disabling window aspect ratio would assert (#852)
- [Cocoa] Bugfix: Window creation failed to set first responder (#876,#883)
- [Cocoa] Bugfix: Removed use of deprecated `CGDisplayIOServicePort` function

`(#165,#192,#508,#511)`

- [Cocoa] Bugfix: Disabled use of deprecated `CGDisplayModeCopyPixelEncoding`

`function on macOS 10.12+`

- [Cocoa] Bugfix: Running in AppSandbox would emit warnings (#816,#882)
- [Cocoa] Bugfix: Windows created after the first were not cascaded (#195)
- [Cocoa] Bugfix: Leaving video mode with `glfwSetWindowMonitor` would set

incorrect position and size (#748)

- [Cocoa] Bugfix: Iconified full screen windows could not be restored (#848)
- [Cocoa] Bugfix: Value range was ignored for joystick hats and buttons (#888)
- [Cocoa] Bugfix: Full screen framebuffer was incorrectly sized for some video

modes (#682)

- [Cocoa] Bugfix: A string object for IME was updated non-idiomatically (#1050)
- [Cocoa] Bugfix: A hidden or disabled cursor would become visible when a user

notification was shown (#971, #1028)

- [Cocoa] Bugfix: Some characters did not repeat due to Press and Hold (#1010)
- [Cocoa] Bugfix: Window title was lost when full screen or undecorated (#1082)
- [Cocoa] Bugfix: Window was resized twice when entering full screen (#1085)
- [Cocoa] Bugfix: Duplicate size events were not filtered (#1085)
- [Cocoa] Bugfix: Event polling did not initialize AppKit if necessary (#1218)
- [Cocoa] Bugfix: OpenGL rendering was not initially visible on 10.14

(#1334, #1346)

- [Cocoa] Bugfix: Caps Lock did not generate any key events (#1368, #1373)
- [Cocoa] Bugfix: Some buttons for some joysticks were ignored (#1385)
- [Cocoa] Bugfix: Analog joystick buttons were not translated correctly (#1385)
- [Cocoa] Bugfix: OpenGL swap interval was ignored for occluded windows (#680)
- [Cocoa] Bugfix: OpenGL swap interval was ignored on macOS 10.14

(#1337, #1417, #1435)

- [WGL] Added support for `WGL_EXT_colorspace` for OpenGL ES contexts
- [WGL] Added support for `WGL_ARB_create_context_no_error`
- [GLX] Added support for `GLX_ARB_create_context_no_error`
- [GLX] Bugfix: Context creation could segfault if no GLXFBConfigs were

available (#1040)

- [EGL] Added support for `EGL_KHR_get_all_proc_addresses` (#871)
- [EGL] Added support for `EGL_KHR_context_flush_control`
- [EGL] Bugfix: The test for `EGL_RGB_BUFFER` was invalid

Contact

On glfw.org you can find the latest version of GLFW, as well as news, documentation and other information about the project.

If you have questions related to the use of GLFW, we have a [forum](#), and the `#glfw` IRC channel on [Freenode](#).

If you have a bug to report, a patch to submit or a feature you'd like to request, please file it in the [issue tracker](#) on GitHub.

Finally, if you're interested in helping out with the development of GLFW or porting it to your favorite platform, join us on the forum, GitHub or IRC.

Acknowledgements

GLFW exists because people around the world donated their time and lent their skills.

- Bobyshev Alexander
- Matt Arsenault
- David Avedissian
- Keith Bauer
- John Bartholomew
- Coşku Baş
- Niklas Behrens
- Andrew Belt
- Niklas Bergström
- Denis Bernard
- Doug Binks
- blanco
- Kyle Brenneman
- Rok Breulj
- Martin Capitanio
- David Carlier
- Arturo Castro
- Chi-kwan Chan
- Ian Clarkson
- Michał Cichoń
- Lambert Clara
- Yaron Cohen-Tal
- Omar Cornut
- Andrew Corrigan
- Bailey Cosier
- Noel Cower
- Jason Daly
- Jarrod Davis
- Olivier Delannoy
- Paul R. Deppe
- Michael Dickens
- Роман Донченко
- Mario Dorn
- Wolfgang Draxinger
- Jonathan Dummer
- Ralph Eastwood
- Fredrik Ehnborn
- Robin Eklind
- Siavash Eliasi
- Felipe Ferreira
- Michael Fogleman
- Gerald Franz
- Mário Freitas
- GeO4d
- Marcus Geelhard
- Stephen Gowen
- Kovid Goyal
- Eloi Marín Gratacós

- Stefan Gustavson
- Jonathan Hale
- Sylvain Hellegouarch
- Matthew Henry
- heromyth
- Lucas Hinderberger
- Paul Holden
- Warren Hu
- IntellectualKitty
- Aaron Jacobs
- Erik S. V. Jansson
- Toni Jovanoski
- Arseny Kapoulkine
- Cem Karan
- Osman Keskin
- Josh Kilmer
- Cameron King
- Peter Knut
- Christoph Kubisch
- Yuri Kunde Schlesner
- Konstantin Käfer
- Eric Larson
- Robin Leffmann
- Glenn Lewis
- Shane Liesegang
- Eyal Lotem
- Aaron Loucks
- Tristram MacDonald
- Hans Mackowiak
- Дмитри Малышев
- Zbigniew Mandziejewicz
- Adam Marcus
- Célestin Marot
- Kyle McDonald
- David Medlock
- Bryce Mehring
- Jonathan Mercier
- Marcel Metz
- Liam Middlebrook
- Jonathan Miller
- Kenneth Miller
- Bruce Mitchener
- Jack Moffitt
- Jeff Molofee
- Alexander Monakov
- Pierre Morel
- Jon Morton
- Pierre Moulon
- Martins Mozeiko
- Julian Møller
- ndogxj
- Kristian Nielsen

- Kamil Nowakowski
- Denis Ovod
- Ozzy
- Andri Pálsson
- Peoro
- Braden Pellett
- Christopher Pelloux
- Arturo J. Pérez
- Anthony Pesch
- Orson Peters
- Emmanuel Gil Peyrot
- Cyril Pichard
- Keith Pitt
- Stanislav Podgorskiy
- Nathan Poirier
- Alexandre Pretymann
- przemekmirek
- Philip Rideout
- Eddie Ringle
- Max Risuhin
- Jorge Rodriguez
- Ed Ropple
- Aleksey Rybalkin
- Riku Salminen
- Brandon Schaefer
- Sebastian Schuberth
- Christian Sdunek
- Matt Sealey
- Steve Sexton
- Arkady Shapkin
- Yoshiki Shibukawa
- Dmitri Shuralyov
- Daniel Skorupski
- Bradley Smith
- Patrick Snape
- Erlend Sogge Heggen
- Julian Squires
- Johannes Stein
- Pontus Stenatorp
- Michael Stocker
- Justin Stoecker
- Elviss Strazdins
- Paul Sultana
- Nathan Sweet
- TTK-Bandit
- Sergey Tikhomirov
- Arthur Tombs
- Ioannis Tsakpinis
- Samuli Tuomola
- Matthew Turner
- urraka
- Elias Vanderstuyft

- Stef Velzel
- Jari Vetoniemi
- Ricardo Vieira
- Nicholas Vitovitch
- Simon Voordouw
- Corentin Wallez
- Torsten Walluhn
- Patrick Walton
- Xo Wang
- Jay Weisskopf
- Frank Wille
- Ryogo Yoshimura
- Andrey Zholos
- Santi Zupancic
- Jonas Ådahl
- Lasse Öörni
- All the unmentioned and anonymous contributors in the GLFW community, for bug reports, patches, feedback, testing and encouragement