

wxWidgets Overview

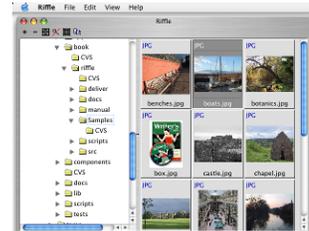
The same source... multiple platforms



Riffle on Windows XP



Riffle on Linux/GTK+



Riffle on Mac OS X



Riffle on Pocket PC 2003

What is wxWidgets?

wxWidgets is a free cross-platform toolkit to help you build desktop and mobile GUI applications on multiple operating systems, from the same source code. It's an open source project, so you can change the wxWidgets source code yourself. Unlike other such toolkits, wxWidgets uses **native widgets** wherever possible, so a wxWidgets application looks and feels as the user expects on each platform. This is a critical requirement for many applications. The principle is demonstrated in the screenshots above: a wxWidgets-based image browser called "Riffle" is running on four platforms.

Although wxWidgets is written in C++, you can use it with a variety of languages including Python, Perl, and C#. If using wxWidgets with C++, you will link your code to a different version of the library on each platform. Since the wxWidgets libraries are built and compiled in C++ rather than a language like Java, they are high-performance and nearly as fast as using the native toolkits themselves.

What platforms does wxWidgets support?

wxWidgets currently has the following stable ports:

- wxGTK** The recommended port for Linux and other Unix variants, using the GTK+ widget set.
- wxMSW** The port for 32-bit and 64-bit Windows variants, including Windows 98, Windows 2000, Windows NT, Windows XP.
- wxWinCE** The port for Windows CE/Windows Mobile based devices, including support for Pocket PC 2003 and Smartphone 2003. This port is contained in the standard wxMSW distribution.
- wxMac** For delivering Carbon applications on Mac OS 9 and Max OS X 10.2 and above.
- wxX11** A port for Linux and Unix variants targeting X11 displays using wxWidgets' own widget set.
- wxMGL** A port for the MGL toolkit from SciTech Software Inc.
- wxMotif** A port for Linux and Unix variants using OpenMotif or Lesstif widget sets.

Other ports in development include **wxCocoa** targeting Mac OS X using the Cocoa toolkit, **wxOS2** targeting OS/2 Presentation Manager, and **wxPalmOS** targeting PalmOS devices.

What functionality does wxWidgets provide?

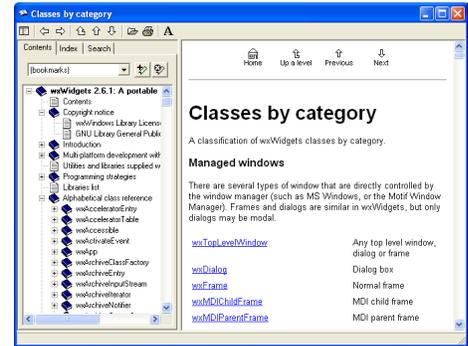
wxWidgets has hundreds of classes covering many areas of application development, which we cannot adequately summarize in this short document. GUI components range from a simple button component to an HTML list box; from a basic message box to a print preview window. Other areas include:

- window layout using sizers;
- device contexts and drawing objects such as pens, brushes and fonts;
- a comprehensive event-handling model;
- sound and video playback;
- Unicode and internationalization support;
- document/view architecture;
- printing architecture and associated dialogs;
- sockets;
- multithreading;
- files, virtual file systems and streams including easy zip archive manipulation;
- online and context-sensitive help;
- HTML rendering, useful for enhanced 'about' boxes, reports and much more;
- containers such as arrays and lists;
- image loading, saving, drawing and manipulation;
- timers and date manipulation;
- error logging;
- clipboard and drag and drop;
- ODBC database access.

You may find it helpful to browse the wxWidgets Reference Manual to get a feel for supported functionality, especially the *Classes By Category* topic.

What is the wxWidgets license?

The wxWidgets library is distributed under the “wxWindows License”, which is based on the L-GPL but with an exception clause. The exception clause allows you to link your application either dynamically or statically to wxWidgets without having to distribute the source for your own application. In other words, you can use wxWidgets for either free or commercial projects, at no cost. The license encourages you to give back enhancements you make to the wxWidgets library itself. For more information, see www.wxwidgets.org/newlicen.htm.



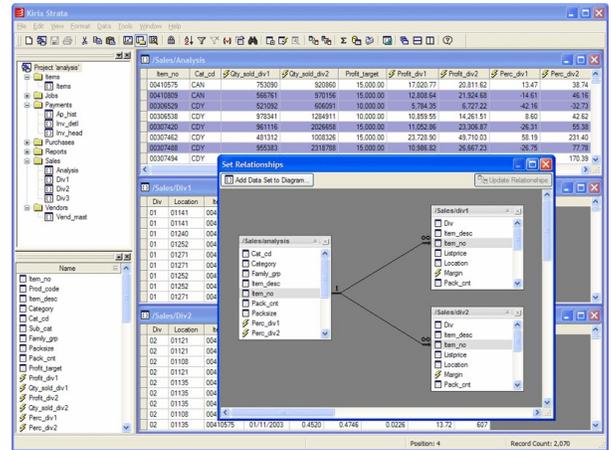
The wxWidgets HTML Help viewer



Audacity, a popular free audio editor

Who uses wxWidgets?

wxWidgets is used by a huge range of organisations and individuals all over the world. It's equally at home being the basis for a consumer product selling hundreds of thousands of copies as it is in university or open source projects. wxWidgets has been used to help companies create leading-edge chips, to help drill for oil, to control pilotless aircraft, and to test components of space telescopes. Many companies are dependent on wxWidgets and the cross-platform advantage it gives them. Some of the better-known organisations who have used wxWidgets include AOL, AMD, Lockheed Martin, Xerox, NASA, and the Open Source Applications Foundation (OSAF). wxWidgets applications that you may be familiar with include AVG AntiVirus, Forte Agent, Audacity, iPodder, and Tortoise CVS. It's impossible to know how many wxWidgets developers there are but there is a very active wxWidgets community with over 1,800 subscribers to the bulletin board alone.



Kirix Strata, winner of the Linuxworld 2005 Product Excellence Award for Best Desktop/Productivity/Business Application

What are the benefits of using wxWidgets?

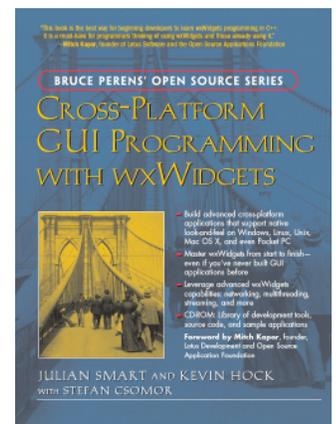
The benefits include the following:

- Cost savings from writing code once that will run on Windows, Unix, Mac OS X, and other platforms.
- Customer satisfaction from delivering stable, fast, attractive applications with a native look and feel.
- Increased productivity from the wide variety of classes that wxWidgets provides, both for creating great GUIs and for general application development.
- Increased market share due to support for platforms you may not have previously considered, and the ability to internationalize your applications.
- Support from a large, active wxWidgets community that answers questions helpfully and provides prompt bug-fixing.
- Access to the source, for enhancement and trouble-shooting.

How do I learn wxWidgets using C++?

When you download wxWidgets, you get a 2000-page reference manual and around 80 samples and demos. This provides a mass of information to help you get started, and you can also buy the 700-page book *Cross-Platform GUI Programming With wxWidgets*, by Julian Smart and Kevin Hock with Stefan Csomor. The dialog editor on the accompanying CD will help you get to grips with sizers, a flexible layout mechanism. The book is available from Amazon and major bookstores; for more information, see www.wxwidgets.org/book/.

When you get stuck, developers and users are available on the forum and wx-users mailing list to answer your questions. Although wxWidgets is a large library, you do not need to learn it all at once, and users have often observed that the API is intuitive and “just makes sense”. If you have used MFC, you will find many familiar concepts.

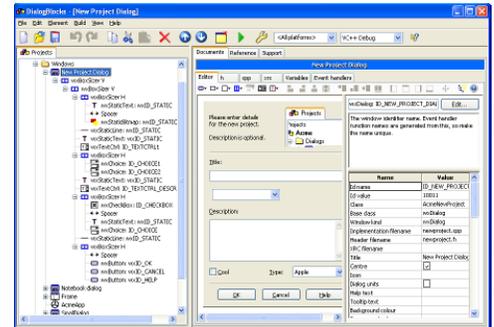


How do I learn wxWidgets using Python?

A wxPython book is currently in development, but in the meantime you can download the wxPython libraries, along with the docs and demos package, from www.wxpython.org. The docs and demos package contains an application showing the use of nearly every class available in wxWidgets, and the demos are dynamically editable so that you can make changes and see the effects in real-time. This is an excellent way to learn how the toolkit works. There is also a Wiki with lots of tips and tricks available at wiki.wxpython.org.

What tools can I use with wxWidgets?

wxWidgets supports most popular compilers, including Microsoft Visual C++, GCC (on Linux, Mac and Windows), Borland C++, Watcom C++, Digital Mars C++, and Sun CC. You can use free or commercial dialog editor and RAD (Rapid Application Development tools), such as wxDesigner and DialogBlocks. There are now several IDEs written in and for wxWidgets, such as CodeBlocks. To create online help for wxWidgets applications, you can use the commercial HelpBlocks application. For more on these and other tools, please see www.wxwidgets.org/lnk_tools.htm.



DialogBlocks , PC Plus Performance Award Winner

For a variety of third-party classes, see www.wxwidgets.org/contrib2.htm and wxcode.sf.org.

What support is available?

Support is via the forum and mailing lists, and commercial support is provided by several companies. See www.wxwidgets.org/support.htm for more on these.

For more information

The web site www.wxwidgets.org should have the answer to most of your questions. If you have further queries, please ask on the mailing list or mail Julian Smart at julian@wxwidgets.org.

"wxWidgets is a very powerful and almost complete toolkit that will reduce development times significantly. Response time of the wxWidgets developers to found bugs is very quick. Active community. It's fantastic that you can get a toolkit like this for free, even for professional use!" *Markus Juergens, Germany*

"I've been looking for a great GUI for C++ for 4 years. I've tried Windows SDK, MFC, Qt, I decided to change to Java, and found out that I didn't like it. Then I found wxWidgets! It is exactly what I have been looking for! In addition, it is cross platform, which is what I wished I could do. Thank you so much for developing this!" *Nuttapong Chentanez, University of Michigan*

"Before wxWidgets I spent a lot of time implementing tools for processing our SGML/XML and illustration documents in C and WIN32. Since using wxWidgets, development is much faster, GUIs are easier to implement and less errors remain hidden... Requests for new functionality in the apps can be satisfied in half the time or less than before. I tried out other frameworks but wxWidgets is the very best I could find suitable for my needs. You all did a very great job! Thank you!" *Armin Kurz, Eurocopter*

"I'd like to thank you for a great library; wxWidgets has made my programming life much easier. MeshCam has been downloaded by several thousand people with no wx-related problems. I'm glad I tried wx and left the world of MFC behind." *Robert, GRZ Software*

"Last thursday I had a call from a very big chemicals company who wanted me to write a Windows application. My only problem was that I had never written a line of code in C, C++ or used the Win32 API before. Thx to wxWidgets I managed to write an OLAP tool using a SQLite database within 3 days! I have never seen an API which was more comfortable to use. Great Job, Thx!" *Ulrich*