wxWidgets Data Sheet
© wxWidgets Development Team, 2005
www.wxwidgets.org

wxWidgets for Embedded Platforms

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.

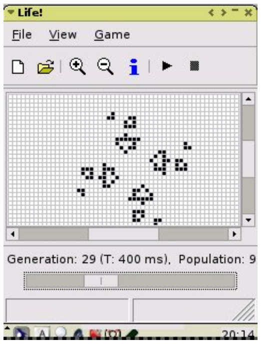
What embedded platforms does wxWidgets support?

The following ports can be used for targetting embedded systems with wxWidgets:

wxGTK: a port for Linux and other Unix variants, using the GTK+ widget set.
wxWinCE: the port for Windows CE/Windows Mobile based devices, including support for Pocket PC 2003 and Smartphone 2003.
wxX11: a port for Linux and Unix variants targetting X11 displays using wxWidgets' own widget set.
wxMGL: a port for the MGL toolkit from SciTech Software Inc.

There is also a **wxPalmOS** port in development, targetting PalmOS devices.

Using wxGTK



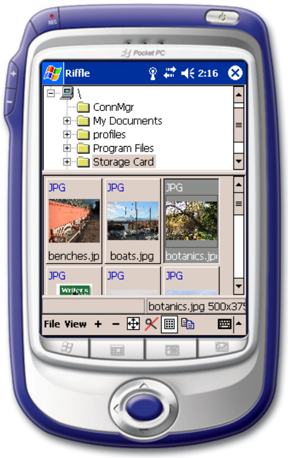
Generation: 29 (T: 400 ms), Population: 9

Since GTK+ 2 can be used on small devices running Linux and a cut-down version of X11, such as Tiny-X, wxGTK is a viable port for developing embedded applications. The screenshot on the right shows the wxWidgets "Life!" demo running under the GPE Palmtop System.

Currently there are no specific adaptations that wxGTK provides for embedded use. In some cases, device-specific additional API may need to be wrapped in wxWidgets-friendly classes. The way in which windows are presented to the user will require design decisions to be made that differ from regular desktop programming; for example, you can expect most frames (top-level windows) to be full-screen.

Using wxWinCE

The Windows CE port can be used with various Windows Mobile SDKs, such as Pocket PC 2003 and Smartphone 2003. At the time of writing, Embedded Visual C++ 4 is the recommended environment to create Windows Mobile applications. The wxWinCE port is included in the standard wxWidgets for Windows distribution, and instructions for use are contained in the document docs/msw/wince/readme.txt and also in the wxMSW section of the wxWidgets Reference Manual.



The wxWinCE port shares much code with the standard Windows port (wxMSW) but there are many adaptations and some limitations. This is a list of some them:

- combined menubars/toolbars, handled automatically by wxWidgets when creating a menubar and toolbar for a frame;
- dialogs on Smartphone have extra functionality for assign labels and commands to left and right menu buttons;
- wxPropertySheetDialog uses a standard Pocket PC property sheet on this platform;
- wxTextCtrl has a wxTE_CAPITALIZE style on Pocket PC;
- wxWinceHelpController is available for controlling simple HTML help files;
- wxCheckListBox, MDI classes and wxMiniFrame are not supported on wxWinCE.

For more detail, please see the wxMSW section in the Reference Manual. The “Life!” demo is optimized for Pocket PC (see demos/life in your distribution), and an additional Pocket PC sample is supplied with the book *Cross-Platform GUI Programming with wxWidgets*.

Using wxX11

wxX11 targets X11 directly, on Unix-based systems, and implements its own widgets. It can be used for embedded systems that have no native widget set, such as custom industrial devices. wxX11 is less well supported than some other ports, and some work may need to be done to optimize wxX11 for the intended applications.

Using wxMGL

wxMGL targets the 2D graphics layer MGL from SciTech Software Inc. Like wxX11, it uses its own set of widgets since MGL does not itself provide widgets. MGL supports Windows, Linux, DOS, OS/2, QNX, SMX, RT-Target and other embedded operating systems, and so wxMGL can be ported to these environments.

wxWidgets API for adapting to small devices

wxWidgets provides a small number of adaptations to deal with small devices in general.

- wxSystemSettings::GetScreenType() returns one of wxSYS_SCREEN_NONE, wxSYS_SCREEN_TINY, wxSYS_SCREEN_PDA, wxSYS_SCREEN_SMALL, and wxSYS_SCREEN_DESKTOP which can be tested at runtime to determine the kind of display the program is running on. For more precise display size values, wxSystemSettings::GetMetric() can be called with wxSYS_SCREEN_X or wxSYS_SCREEN_Y.
- For testing at compile type, you can use symbols such as __PDA__ and __SMARTPHONE__.
- Since notebook (tab control) classes can take up a lot of valuable space, the class wxBookCtrl is provided that uses a drop-down choice control on smart phone platforms, and is implemented by a wxNotebook on other platforms.