

NAME

perlce - Perl for WinCE

DESCRIPTION

This file gives the instructions for building Perl5.8 and above for WinCE. Please read and understand the terms under which this software is distributed.

BUILD

This section describes the steps to be performed to build PerlCE. You may find additional and newer information about building perl for WinCE using following URL:

`http://perlce.sourceforge.net`

There should also be pre-built binaries there.

Don't be confused by large size of downloaded distribution or constructed binaries: entire distribution could be large for WinCE ideology, but you may strip it at your wish and use only required parts.

Tools & SDK

For compiling, you need following:

- * Microsoft Embedded Visual Tools
- * Microsoft Visual C++
- * Rainer Keuchel's celib-sources
- * Rainer Keuchel's console-sources

Needed source files can be downloaded via: www.rainer-keuchel.de/wince/dirlist.html

Make

Please pay attention that starting from 5.8.0 miniperl *is* built and it facilitates in further building process. This means that in addition to compiler installation for mobile device you also need to have Microsoft Visual C++ installed as well.

On the bright side, you do not need to edit any files from `./win32` subdirectory. Normally you only need to edit `./wince/compile.bat` to reflect your system and run it.

File `./wince/compile.bat` is actually a wrapper to call `nmake -f makefile.ce` with appropriate parameters and it accepts extra parameters and forwards them to "nmake" command as additional arguments. You should pass target this way.

To prepare distribution you need to do following:

- * go to `./wince` subdirectory
- * edit file `compile.bat`
- * run `compile.bat`
- * run `compile.bat dist`

`makefile.ce` has `CROSS_NAME` macro, and it is used further to refer to your cross-compilation scheme. You could assign a name to it, but this is not necessary, because by default it is assigned after your machine configuration name, such as `"wince-sh3-hpc-wce211"`, and this is enough to distinguish different builds at the same time. This option could be handy for several different builds on same platform to perform, say, threaded build. In a following example we assume that all required environment variables are set properly for C cross-compiler (a special `*.bat` file could fit perfectly to this purpose) and your `compile.bat` has proper "MACHINE" parameter set, to, say, `"wince-mips-pocket-wce300"`.

```
compile.bat
```

```
compile.bat dist
compile.bat CROSS_NAME=mips-wce300-thr "USE_ITHREADS=define"
"USE_IMP_SYS=define" "USE_MULTI=define"
compile.bat CROSS_NAME=mips-wce300-thr "USE_ITHREADS=define"
"USE_IMP_SYS=define" "USE_MULTI=define" dist
```

If all goes okay and no errors during a build, you'll get two independent distributions: "wince-mips-pocket-wce300" and "mips-wce300-thr".

Target 'dist' prepares distribution file set. Target 'zipdist' performs same as 'dist' but additionally compresses distribution files into zip archive.

NOTE: during a build there could be created a number (or one) of Config.pm for cross-compilation ("foreign" Config.pm) and those are hidden inside ../xlib/\$(CROSS_NAME) with other auxiliary files, but, and this is important to note, there should be **no** Config.pm for host miniperl. If you'll get an error that perl could not find Config.pm somewhere in building process this means something went wrong. Most probably you forgot to specify a cross-compilation when invoking miniperl.exe to Makefile.PL When building an extension for cross-compilation your command line should look like

```
..\miniperl.exe -I..\lib -MCross=mips-wce300-thr Makefile.PL
```

or just

```
..\miniperl.exe -I..\lib -MCross Makefile.PL
```

to refer a cross-compilation that was created last time.

If you decided to build with fcrypt.c file, please refer to README.win32 file, as long as all legal considerations and steps to do are exactly same in this case.

All questions related to building for WinCE devices could be asked in perlce-users@lists.sourceforge.net mailing list.

ACKNOWLEDGEMENTS

The port for Win32 was used as a reference.

AUTHORS

Rainer Keuchel (keuchel@netwave.de) Vadim Konovalov (vkonovalov@spb.lucent.com)