

NAME

perlsymbian - Perl version 5 on Symbian OS

DESCRIPTION

This document describes various features of the Symbian operating system that will affect how Perl version 5 (hereafter just Perl) is compiled and/or runs.

NOTE: this port (as of 0.4.1) does not compile into a Symbian OS GUI application, but instead it results in a Symbian DLL. The DLL includes a C++ class called CPerlBase, which one can then (derive from and) use to embed Perl into applications, see *symbian/README*.

The base port of Perl to Symbian only implements the basic POSIX-like functionality; it does not implement any further Symbian or Series 60, Series 80, or UIQ bindings for Perl.

It is also possible to generate Symbian executables for “miniperl” and “perl”, but since there is no standard command line interface for Symbian (nor full keyboards in the devices), these are useful mainly as demonstrations.

Compiling Perl on Symbian

(0) You need to have the appropriate Symbian SDK installed.

These instructions have been tested under various Nokia Series 60 Symbian SDKs (1.2 to 2.6, 2.8 should also work, 1.2 compiles but does not work), Series 80 2.0, and Nokia 7710 (Series 90) SDK. You can get the SDKs from Forum Nokia (<<http://www.forum.nokia.com/>>). A very rough port (“it compiles”) to UIQ 2.1 has also been made.

A prerequisite for any of the SDKs is to install ActivePerl from ActiveState, <<http://www.activestate.com/Products/ActivePerl/>>

Having the SDK installed also means that you need to have either the Metrowerks CodeWarrior installed (2.8 and 3.0 were used in testing) or the Microsoft Visual C++ 6.0 installed (SP3 minimum, SP5 recommended).

Note that for example the Series 60 2.0 VC SDK installation talks about ActivePerl build 518, which does no more (as of mid-2005) exist at the ActiveState website. The ActivePerl 5.8.4 build 810 was used successfully for compiling Perl on Symbian. The 5.6.x ActivePerls do not work.

Other SDKs or compilers like Visual.NET, command-line-only Visual.NET, Borland, GnuPoc, or sdk2unix have not been tried.

These instructions almost certainly won’t work with older Symbian releases or other SDKs. Patches to get this port running in other releases, SDKs, compilers, platforms, or devices are naturally welcome.

(1) Get a Perl source code distribution (for example the file perl-5.9.2.tar.gz is fine) from <<http://www.cpan.org/src/>> and unpack it in your the C:/Symbian directory of your Windows system.

(2) Change to the perl source directory.

```
cd c:\Symbian\perl-5.x.x
```

(3) Run the following script using the perl coming with the SDK

```
perl symbian\config.pl
```

You must use the cmd.exe, the Cygwin shell will not work. The PATH must include the SDK tools, including a Perl, which should be the case under cmd.exe. If you do not have that, see the end of symbian\sdk.pl for notes of how your environment should be set up for Symbian compiles.

(4) Build the project, either by

```
make all
```

in cmd.exe or by using either the Metrowerks CodeWarrior or the Visual C++ 6.0, or the Visual Studio 8 (the Visual C++ 2005 Express Edition works fine).

If you use the VC IDE, you will have to run *symbian\config.pl* first using the cmd.exe, and then run ‘make win.mf vc6.mf’ to generate the VC6 makefiles and workspaces. “make vc6” will compile for the VC6, and

“make cw” for the CodeWarrior.

The following SDK and compiler configurations and Nokia phones were tested at some point in time (+ = compiled and PerlApp run, - = not), both for Perl 5.8.x and 5.9.x:

SDK	VC	CW	
S60 1.2	+	+	3650 (*)
S60 2.0	+	+	6600
S60 2.1	-	+	6670
S60 2.6	+	+	6630
S60 2.8	+	+	(not tested in a device)
S80 2.6	-	+	9300
S90 1.1	+	-	7710
UIQ 2.1	-	+	(not tested in a device)

(*) Compiles but does not work, unfortunately, a problem with Symbian.

If you are using the 'make' directly, it is the GNU make from the SDKs, and it will invoke the right make commands for the Windows emulator build and the Arm target builds ('thumb' by default) as necessary.

The build scripts assume the 'absolute style' SDK installs under C:, the 'subst style' will not work.

If using the VC IDE, to build use for example the File->Open Workspace-> C:\Symbian\8.0a\S60_2nd_FP2\epoc32\build\symbian\perl\perl\wins\perl.dsw The emulator binaries will appear in the same directory.

If using the VC IDE, you will a lot of warnings in the beginning of the build because a lot of headers mentioned by the source cannot be found, but this is not serious since those headers are not used.

The Metrowerks will give a lot of warnings about unused variables and empty declarations, you can ignore those.

When the Windows and Arm DLLs are built do not be scared by a very long messages whizzing by: it is the “export freeze” phase where the whole (rather large) API of Perl is listed.

Once the build is completed you need to create the DLL SIS file by

```
make perldll.sis
```

which will create the file perlXYZ.sis (the XYZ being the Perl version) which you can then install into your Symbian device: an easy way to do this is to send them via Bluetooth or infrared and just open the messages.

Since the total size of all Perl SIS files once installed is over 2 MB, it is recommended to do the installation into a memory card (drive E:) instead of the C: drive.

The size of the perlXYZ.SIS is about 370 kB but once it is in the device it is about one 750 kB (according to the application manager).

The perlXYZ.sis includes only the Perl DLL: to create an additional SIS file which includes some of the standard (pure) Perl libraries, issue the command

```
make perllib.sis
```

Some of the standard Perl libraries are included, but not all: see “HISTORY” or *symbian\install.cfg* for more details (250 kB -> 700 kB).

Some of the standard Perl XS extensions (see “HISTORY” are also available:

```
make perlext.sis
```

which will create perlXYZext.sis (290 kB -> 770 kB).

To compile the demonstration application PerlApp you need first to install the Perl headers under the SDK.

To install the Perl headers and the class CPerlBase documentation so that you no more need the Perl

sources around to compile Perl applications using the SDK:

```
make sdkinstall
```

The destination directory is C:\Symbian\perl\X.Y.Z. For more details, see *symbian\PerlBase.pod*.

Once the headers have been installed, you can create a SIS for the PerlApp:

```
make perlapp.sis
```

The perlapp.sis (11 kB -> 16 kB) will be built in the symbian subdirectory, but a copy will also be made to the main directory.

If you want to package the Perl DLLs (one for WINS, one for ARMI), the headers, and the documentation:

```
make perlsdk.zip
```

which will create perlXYZsdk.zip that can be used in another Windows system with the SDK, without having to compile Perl in that system.

If you want to package the PerlApp sources:

```
make perlapp.zip
```

If you want to package the perl.exe and miniperl.exe, you can use the perlexe.sis and miniperlexe.sis make targets. You also probably want the perllib.sis for the libraries and maybe even the perlapp.sis for the recognizer.

The make target 'allsis' combines all the above SIS targets.

To clean up after compilation you can use either of

```
make clean
make distclean
```

depending on how clean you want to be.

Compilation problems

If you see right after “make” this

```
cat makefile.sh >makefile
'cat' is not recognized as an internal or external command,
operable program or batch file.
```

it means you need to (re)run the *symbian\config.pl*.

If you get the error

```
'perl' is not recognized as an internal or external command,
operable program or batch file.
```

you may need to reinstall the ActivePerl.

If you see this

```
ren makedef.pl nomakedef.pl
The system cannot find the file specified.
C:\Symbian\...\make.exe: [rename_makedef] Error 1 (ignored)
```

please ignore it since it is nothing serious (the build process of renames the Perl makedef.pl as nomakedef.pl to avoid confusing it with a makedef.pl of the SDK).

PerlApp

The PerlApp application demonstrates how to embed Perl interpreters to a Symbian application. The “Time” menu item runs the following Perl code: `print "Running in ", $^O, "\n", scalar localtime,` the “Oneliner” allows one to type in Perl code, and the “Run” opens a file chooser for selecting a Perl file to run.

The PerlApp also is started when the “Perl recognizer” (also included and installed) detects a Perl file being activated through the GUI, and offers either to install it under \Perl (if the Perl file is in the inbox of

the messaging application) or to run it (if the Perl file is under \Perl).

sisify.pl

In the symbian subdirectory there is *sisify.pl* utility which can be used to package Perl scripts and/or Perl library directories into SIS files, which can be installed to the device. To run the *sisify.pl* utility, you will need to have the 'makesis' and 'uidcrc' utilities already installed. If you don't have the Win32 SDKs, you may try for example <<http://gnupoc.sourceforge.net/>> or <<http://symbianos.org/~andreh/>>.

Using Perl in Symbian

First of all note that you have full access to the Symbian device when using Perl: you can do a lot of damage to your device (like removing system files) unless you are careful. Please do take backups before doing anything.

The Perl port has been done for the most part using the Symbian standard POSIX-ish STDLIB library. It is a reasonably complete library, but certain corners of such emulation libraries that tend to be left unimplemented on non-UNIX platforms have been left unimplemented also this time: *fork()*, *signals()*, user/group ids, *select()* working for sockets, non-blocking sockets, and so forth. See the file *symbian/config.sh* and look for 'undef' to find the unsupported APIs (or from Perl use Config).

The filesystem of Symbian devices uses DOSish syntax, "drives" separated from paths by a colon, and backslashes for the path. The exact assignment of the drives probably varies between platforms, but for example in Series 60 you might see C: as the (flash) main memory, D: as the RAM drive, E: as the memory card (MMC), Z: as the ROM. In Series 80 D: is the memory card. As far the devices go the NUL: is the bit bucket, the COMx: are the serial lines, IRCOMx: are the IR ports, TMP: might be C:\System\Temp. Remember to double those backslashes in doublequoted strings.

The Perl DLL is installed in \System\Libs\. The Perl libraries and extension DLLs are installed in \System\Libs\Perl\X.Y.Z\. The PerlApp is installed in \System\Apps\, and the SIS also installs a couple of demo scripts in \Perl\ (C:\Mydocs\Perl\ on Nokia 7710).

Note that the Symbian filesystem is very picky: it strongly prefers the \ instead of the /.

When doing XS / Symbian C++ programming include first the Symbian headers, then any standard C/POSIX headers, then Perl headers, and finally any application headers.

New() and *Copy()* are unfortunately used by both Symbian and Perl code so you'll have to play cpp games if you need them. PerlBase.h undefines the Perl definitions and redefines them as *PerlNew()* and *PerlCopy()*.

TO DO

Lots. See *symbian/TODO*.

WARNING

As of Perl Symbian port version 0.4.1 any part of Perl's standard regression test suite has not been run on a real Symbian device using the ported Perl, so innumerable bugs may lie in wait. Therefore there is absolutely no warranty.

NOTE

When creating and extending application programming interfaces (APIs) for Symbian or Series 60 or Series 80 or Series 90 it is suggested that trademarks, registered trademarks, or trade names are not used in the API names. Instead, developers should consider basing the API naming in the existing (C++, or maybe Java) public component and API naming, modified as appropriate by the rules of the programming language the new APIs are for.

Nokia is a registered trademark of Nokia Corporation. Nokia's product names are trademarks or registered trademarks of Nokia. Other product and company names mentioned herein may be trademarks or trade names of their respective owners.

AUTHOR

Jarkko Hietaniemi

COPYRIGHT

Copyright (c) 2004-2005 Nokia. All rights reserved.

Copyright (c) 2006-2007 Jarkko Hietaniemi.

LICENSE

The Symbian port is licensed under the same terms as Perl itself.

HISTORY

- 0.1.0: April 2005

(This will show as "0.01" in the Symbian Installer.)

- The console window is a very simple console indeed: one can get the newline with "000" and the "C" button is a backspace. Do not expect a terminal capable of vt100 or ANSI sequences. The console is also "ASCII", you cannot input e.g. any accented letters. Because of obvious physical constraints the console is also very small: (in Nokia 6600) 22 columns, 17 rows.
- The following libraries are available:
AnyDBM_File AutoLoader base Carp Config Cwd constant
DynaLoader Exporter File::Spec integer lib strict Symbol
vars warnings XSLoader
- The following extensions are available:
attributes Compress::Zlib Cwd Data::Dumper Devel::Peek
Digest::MD5 DynaLoader Fcntl File::Glob Filter::Util::Call
IO List::Util MIME::Base64
PerlIO::scalar PerlIO::via SDBM_File Socket Storable Time::HiRes
- The following extensions are missing for various technical reasons:
B ByteLoader Devel::DProf Devel::PPPort Encode GDBM_File
I18N::Langinfo IPC::SysV NDBM_File Opcode PerlIO::encoding POSIX
re Safe Sys::Hostname Sys::Syslog
threads threads::shared Unicode::Normalize
- Using MakeMaker or the Module::* to build and install modules is not supported.
- Building XS other than the ones in the core is not supported.

Since this is 0.something release, any future releases are almost guaranteed to be binary incompatible. As a sign of this the Symbian symbol exports are kept unfrozen and the .def files fully rebuilt every time.

- 0.2.0: October 2005
 - Perl 5.9.3 (patch level 25741)
 - Compress::Zlib and IO::Zlib supported
 - sisify.pl added

We maintain the binary incompatibility.

- 0.3.0: October 2005
 - Perl 5.9.3 (patch level 25911)
 - Series 80 2.0 and UIQ 2.1 support

We maintain the binary incompatibility.

- 0.4.0: November 2005
 - Perl 5.9.3 (patch level 26052)
 - adding a sample Symbian extension

We maintain the binary incompatibility.

- 0.4.1: December 2006
 - Perl 5.9.5-to-be (patch level 30002)
 - added extensions: Compress/Raw/Zlib, Digest/SHA, Hash/Util, Math/BigInt/FastCalc, Text/Soundex, Time/Piece
 - port to S90 1.1 by alexander smishlajev

We maintain the binary incompatibility.

- 0.4.2: March 2007
 - catchup with Perl 5.9.5-to-be (patch level 30812)
 - tested to build with Microsoft Visual C++ 2005 Express Edition (which uses Microsoft Visual C 8, instead of the old VC6), SDK used for testing S60_2nd_FP3 aka 8.1a

We maintain the binary incompatibility.