

NAME

perlbeos - Perl version 5.8+ on BeOS

DESCRIPTION

This file contains instructions how to build Perl under BeOS and lists known problems.

BUILD AND INSTALL

Requirements

I have built and tested Perl 5.8.6 and 5.9.1 under BeOS R5 x86 net server. I can't say anything with regard to PPC. Since Perl 5.8.0 had been released for BeOS BONE, I suspect, there is a good chance, that it still compiles on a BONE system. The only change I've made, that affects BONE systems is the recognition of whether it is a BONE system or not in *hints/beos.sh*. Now network socket support should remain enabled on BONE systems. This might as well break the build, though.

As more recent versions of autoconf require flock() support, I wrote a flock() emulation (flock_server) and released it on BeBits:

<http://www.bebits.com/app/4030>

If you want to build a Perl with flock() support, you have to install this package first.

Configure

With flock() support:

```
CFLAGS=-I/path/to/flock/server/headers ./configure.gnu \  
--prefix=/boot/home/config
```

Replace */path/to/flock/server/headers* with the path to the directory containing the *flock.h* header.

Without flock() support:

```
./configure.gnu --prefix=/boot/home/config
```

Build

With flock() support:

```
make LDLOADLIBS="-lnet -lflock"
```

Without flock() support:

```
make LDLOADLIBS="-lnet"
```

-lnet is needed on net server systems only and if the compiler doesn't add it automatically (Be's R5 gcc does, Oliver Tappe's gcc 2.95.3 does not).

Install

Install all perl files:

```
make install
```

Create a symlink for libperl:

```
cd ~/config/lib; ln -s perl5/5.8.6/BePC-beos/CORE/libperl.so .
```

Replace 5.8.6 with your respective version of Perl.

KNOWN PROBLEMS

- Network socket support is disabled for BeOS R5 net server. I didn't dare yet to try enabling it and see what problems occur.
- The LFS (large file support) tests (*t/op/lfs* and *xt/Fcntl/t/syslfs*) are disabled as seeking beyond 2 GB is broken according to jhi@iki.fi who was the last one checking the BeOS port and updating this file before me. Haven't checked this myself.
- The *t/io/flush* test fails at #6. As far as I can tell, this is caused by a bug in the BeOS pipes implementation that occurs when starting other child processes. In the particular test case a `system("perl -e 0")` flushes the stdout pipe of another child process.
- The *ext/POSIX/t/waitpid* test fails at #1. After all child processes are gone BeOS' `waitpid(-1,...)` returns 0 instead of -1 (as it should). No idea how to fix this.

CONTACT

For BeOS specifics problems feel free to mail to: Ingo Weinhold <bonefish@cs.tu-berlin.de>

Last update: 2004-12-16