NAME

README.machten - Perl version 5 on Power MachTen systems

DESCRIPTION

This document describes how to build Perl 5 on Power MachTen systems, and discusses a few wrinkles in the implementation.

Perl version 5.8.x and greater not supported

Power MachTen is not supported by versions of Perl later than 5.6.x. If you wish to build a version from the 5.6 track, please obtain a source distribution from the archive at *http://cpan.org/src/5.0/* and follow the instructions in its README.machten file.

MachTen is no longer supported by its developers, Tenon Intersystems. A UNIX environment hosted on Mac OS Classic, MachTen has been superseded by Mac OS X and by BSD and Linux implementations for Macintosh hardware. The final version of Power MachTen, 4.1.4, lacks many features found in modern implementations of UNIX, and has a number of bugs. These shortcomings prevent recent versions of Perl from being able to use extensions on MachTen, and cause numerous test suite failures in the perl core.

In September 2003, a discussion on the MachTen mailing list determined that there was no interest in making a later version of Perl build successfully on MachTen. Consequently, support for building Perl under MachTen has been suppressed in Perl distributions published after February 2004. The hints file, *hints/machten.sh*, remains a part of the distributions for reference purposes.

Compiling Perl 5.6.x on MachTen

To compile perl 5.6.x under MachTen 4.1.4 (and probably earlier versions):

```
./Configure -de
make
make test
make install
```

This builds and installs a statically-linked perl; MachTen's dynamic linking facilities are not adequate to support Perl's use of dynamically linked libraries. (See *hints/machten.sh* for more information.)

You should have at least 32 megabytes of free memory on your system before running the `make` command.

For much more information on building perl -- for example, on how to change the default installation directory -- see *INSTALL*.

Failures during make test on MachTen

op/lexassign.t

> This test may fail when first run after building perl. It does not fail subsequently. The cause is unknown.

pragma/warnings.t

> Test 257 fails due to a failure to warn about attempts to read from a filehandle which is a duplicate of stdout when stdout is attached to a pipe. The output of the test contains a block comment which discusses a different failure, not applicable to MachTen.

> The root of the problem is that Machten does not assign a file type to either end of a pipe (see *stat*), resulting, among other things in Perl's -p test failing on file descriptors belonging to pipes. As a result, perl becomes confused, and the test for reading from a write-only file fails. I am reluctant to patch perl to get around this, as it's clearly an OS bug (about which Tenon has been informed), and limited in its effect on practical Perl programs.

Building external modules on MachTen

To add an external module to perl, build in the normal way, which is documented in *ExtUtils::MakeMaker*, or which can be driven automatically by the CPAN module (see *CPAN*), which is part of the standard distribution. If you want to install a module which contains XS code (C or C++ source which compiles to object code for linking with perl), you will have to replace your perl binary with a new version containing the new statically-linked object module. The build process tells you how to do this.

There is a gotcha, however, which users usually encounter immediately they respond to CPAN's invitation to `install Bundle::CPAN`. When installing a *bundle* -- a group of modules which together achieve some particular purpose, the installation process for later modules in the bundle tends to assume that earlier modules have been fully installed and are available for use. This is not true on a statically-linked system for earlier modules which contain XS code. As a result the installation of the bundle fails. The work-around is not to install the bundle as a one-shot operation, but instead to see what modules it contains, and install these one-at-a-time by hand in the order given.

AUTHOR

Dominic Dunlop <domo@computer.org>

DATE

Version 1.1.0 2004-02-13