NAME

IO::Tty − Low−level allocate a pseudo−Tty, import constants.

VERSION

1.08

SYNOPSIS

```
    use IO::Tty qw(TIOCNOTTY);
    ...
    # use only to import constants, see IO::Pty to create ptys.
```

DESCRIPTION

`IO::Tty` is used internally by `IO::Pty` to create a pseudo-tty. You wouldn't want to use it directly except to import constants, use `IO::Pty`. For a list of importable constants, see IO::Tty::Constant.

Windows is now supported, but ONLY under the Cygwin environment, see <http://sources.redhat.com/cygwin/>.

Please note that pty creation is very system-dependend. From my experience, any modern POSIX system should be fine. Find below a list of systems that `IO::Tty` should work on. A more detailed table (which is slowly getting out-of-date) is available from the project pages document manager at SourceForge <http://sourceforge.net/projects/expectperl/>.

If you have problems on your system and your system is listed in the "verified" list, you probably have some non-standard setup, e.g. you compiled your Linux-kernel yourself and disabled ptys (bummer!). Please ask your friendly sysadmin for help.

If your system is not listed, unpack the latest version of `IO::Tty`, do a `'perl Makefile.PL; make; make test; uname −a'` and send me (*RGiersig@cpan.org*) the results and I'll see what I can deduce from that. There are chances that it will work right out-of-the-box...

If it's working on your system, please send me a short note with details (version number, distribution, etc. 'uname −a' and 'perl −V' is a good start; also, the output from "perl Makefile.PL" contains a lot of interesting info, so please include that as well) so I can get an overview. Thanks!

VERIFIED SYSTEMS, KNOWN ISSUES

This is a list of systems that `IO::Tty` seems to work on ('make test' passes) with comments about "features":

- AIX 4.3

  Returns EIO instead of EOF when the slave is closed. Benign.

- AIX 5.x
- FreeBSD 4.4

  EOF on the slave tty is not reported back to the master.

- OpenBSD 2.8

  The ioctl TIOCSCTTY sometimes fails. This is also known in Tcl/Expect, see http://expect.nist.gov/FAQ.html

  EOF on the slave tty is not reported back to the master.

- Darwin 7.9.0
- HPUX 10.20 & 11.00

  EOF on the slave tty is not reported back to the master.

- IRIX 6.5
- Linux 2.2.x & 2.4.x

  Returns EIO instead of EOF when the slave is closed. Benign.

- OSF 4.0

  EOF on the slave tty is not reported back to the master.

- Solaris 8, 2.7, 2.6

  Has the "feature" of returning EOF just once?!

  EOF on the slave tty is not reported back to the master.

- Windows NT/2k/XP (under Cygwin)

  When you send (print) a too long line (>160 chars) to a non-raw pty, the call just hangs forever and even **alarm()** cannot get you out. Don't complain to me...

  EOF on the slave tty is not reported back to the master.

- z/OS

The following systems have not been verified yet for this version, but a previous version worked on them:

- SCO Unix
- NetBSD

  probably the same as the other *BSDs...

If you have additions to these lists, please mail them to <*RGiersig@cpan.org*>.

SEE ALSO

IO::Pty, IO::Tty::Constant

MAILING LISTS

As this module is mainly used by Expect, support for it is available via the two Expect mailing lists, expectperl-announce and expectperl-discuss, at

```
  http://lists.sourceforge.net/lists/listinfo/expectperl-announce
```

and

```
  http://lists.sourceforge.net/lists/listinfo/expectperl-discuss
```

AUTHORS

Originally by Graham Barr <*gbarr@pobox.com*>, based on the Ptty module by Nick Ing-Simmons <*nik@tiuk.ti.com*>.

Now maintained and heavily rewritten by Roland Giersig <*RGiersig@cpan.org*>.

Contains copyrighted stuff from openssh v3.0p1, authored by Tatu Ylonen <ylo@cs.hut.fi>, Markus Friedl and Todd C. Miller <Todd.Miller@courtesan.com>. I also got a lot of inspiry from the pty code in Xemacs.

COPYRIGHT

Now all code is free software; you can redistribute it and/or modify it under the same terms as Perl itself.

Nevertheless the above AUTHORS retain their copyrights to the various parts and want to receive credit if their source code is used. See the source for details.

DISCLAIMER

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

In other words: Use at your own risk. Provided as is. Your mileage may vary. Read the source, Luke!

And finally, just to be sure:

Any Use of This Product, in Any Manner Whatsoever, Will Increase the Amount of Disorder in the Universe. Although No Liability Is Implied Herein, the Consumer Is Warned That This Process Will Ultimately Lead to the Heat Death of the Universe.