NAME

prove -- A command-line tool for running tests against Test::Harness

SYNOPSIS

prove [options] [files/directories]

OPTIONS

```
    -b, --blib      Adds blib/lib to the path for your tests, a la "use
blib"
    -d, --debug     Includes extra debugging information
    -D, --dry       Dry run: Show the tests to run, but don't run them
    -h, --help      Display this help
    -H, --man       Longer manpage for prove
    -I              Add libraries to @INC, as Perl's -I
    -l, --lib       Add lib to the path for your tests
        --perl      Sets the name of the Perl executable to use
    -r, --recurse   Recursively descend into directories
    -s, --shuffle   Run the tests in a random order
        --strap     Define strap class to use
    -T              Enable tainting checks
    -t              Enable tainting warnings
        --timer     Print elapsed time after each test file
    -v, --verbose   Display standard output of test scripts while running
them
    -V, --version   Display version info
```

Single-character options may be stacked. Default options may be set by specifying the PROVE_SWITCHES environment variable.

OVERVIEW

prove is a command-line interface to the test-running functionality of `Test::Harness`. With no arguments, it will run all tests in the current directory.

Shell metacharacters may be used with command lines options and will be exanded via `File::Glob::bsd_glob`.

PROVE VS. "MAKE TEST"

prove has a number of advantages over `make test` when doing development.

* *prove* is designed as a development tool

> Perl users typically run the test harness through a makefile via `make test`. That's fine for module distributions, but it's suboptimal for a test/code/debug development cycle.

* *prove* is granular

> *prove* lets your run against only the files you want to check. Running `prove t/live/ t/master.t` checks every **.t* in *t/live*, plus *t/master.t*.

* *prove* has an easy verbose mode

> *prove* has a `-v` option to see the raw output from the tests. To do this with `make test`, you must set `HARNESS_VERBOSE=1` in the environment.

* *prove* can run under taint mode

> *prove*'s `-T` runs your tests under `perl -T`, and `-t` runs them under `perl -t`.

* *prove* can shuffle tests

> You can use *prove*'s `--shuffle` option to try to excite problems that don't show up when

tests are run in the same order every time.

* *prove* doesn't rely on a make tool

Not everyone wants to write a makefile, or use *ExtUtils::MakeMaker* to do so. *prove* has no external dependencies.

* Not everything is a module

More and more users are using Perl's testing tools outside the context of a module distribution, and may not even use a makefile at all.

COMMAND LINE OPTIONS

-b, --blib

Adds blib/lib to the path for your tests, a la "use blib".

-d, --debug

Include debug information about how *prove* is being run. This option doesn't show the output from the test scripts. That's handled by -v,--verbose.

-D, --dry

Dry run: Show the tests to run, but don't run them.

-I

Add libraries to @INC, as Perl's -I.

-l, --lib

Add `lib` to @INC. Equivalent to `-Ilib`.

--perl

Sets the `HARNESS_PERL` environment variable, which controls what Perl executable will run the tests.

-r, --recurse

Descends into subdirectories of any directories specified, looking for tests.

-s, --shuffle

Sometimes tests are accidentally dependent on tests that have been run before. This switch will shuffle the tests to be run prior to running them, thus ensuring that hidden dependencies in the test order are likely to be revealed. The author hopes the run the algorithm on the preceding sentence to see if he can produce something slightly less awkward.

--strap

Sets the HARNESS_STRAP_CLASS variable to set which Test::Harness::Straps variable to use in running the tests.

-t

Runs test programs under perl's -t taint warning mode.

-T

Runs test programs under perl's -T taint mode.

--timer

Print elapsed time after each test file

-v, --verbose

Display standard output of test scripts while running them. Also sets TEST_VERBOSE in case your tests rely on them.

-V, --version

>   Display version info.

BUGS

>   Please use the CPAN bug ticketing system at *http://rt.cpan.org/*. You can also mail bugs, fixes and enhancements to `<bug-test-harness@rt.cpan.org>`.

TODO

>   ● Shuffled tests must be recreatable

AUTHORS

>   Andy Lester `<andy at petdance.com>`

COPYRIGHT

>   Copyright 2004-2006 by Andy Lester `<andy at petdance.com>`.

>   This program is free software; you can redistribute it and/or modify it under the same terms as Perl itself.

>   See *http://www.perl.com/perl/misc/Artistic.html*.