

NAME

TAP::Harness::Env - Parsing harness related environmental variables where appropriate

VERSION

Version 3.36

SYNOPSIS

```
my $harness = TAP::Harness::Env->create(\%extra_args)
```

DESCRIPTION

This module implements the environmental variables that [Test::Harness](#) uses with [TAP::Harness](#), and instantiates the appropriate class with the appropriate arguments.

METHODS

- `create(\%args)`

This function reads the environment and generates an appropriate argument hash from it. If given any arguments in `%extra_args`, these will override the environmental defaults. It accepts `harness_class` (which defaults to [TAP::Harness](#) and any argument the harness class accepts.

ENVIRONMENTAL VARIABLES**HARNESS_PERL_SWITCHES**

Setting this adds perl command line switches to each test file run.

For example, `HARNESS_PERL_SWITCHES=-T` will turn on taint mode. `HARNESS_PERL_SWITCHES=-MDevel::Cover` will run `Devel::Cover` for each test.

HARNESS_VERBOSE

If true, [TAP::Harness](#) will output the verbose results of running its tests.

HARNESS_SUBCLASS

Specifies a [TAP::Harness](#) subclass to be used in place of [TAP::Harness](#).

HARNESS_OPTIONS

Provide additional options to the harness. Currently supported options are:

`j<n>`

Run `<n>` (default 9) parallel jobs.

`c` Try to color output. See “new” in [TAP::Formatter::Base](#).

`a<file.tgz>`

Will use [TAP::Harness::Archive](#) as the harness class, and save the TAP to `file.tgz`

`fPackage-With-Dashes`

Set the `formatter_class` of the harness being run. Since the `HARNESS_OPTIONS` is separated by `:`, we use `-` instead.

Multiple options may be separated by colons:

```
HARNESS_OPTIONS=j9:c make test
```

HARNESS_TIMER

Setting this to true will make the harness display the number of milliseconds each test took. You can also use *prove*'s `--timer` switch.

HARNESS_COLOR

Attempt to produce color output.

HARNESS_IGNORE_EXIT

If set to a true value instruct [TAP::Parser](#) to ignore exit and wait status from test scripts.