

## NAME

Module::Build::Platform::VMS - Builder class for VMS platforms

## DESCRIPTION

This module inherits from `Module::Build::Base` and alters a few minor details of its functionality. Please see *Module::Build* for the general docs.

## Overridden Methods

### `_set_defaults`

Change `$self->{build_script}` to 'Build.com' so `@Build` works.

### `cull_args`

'`@Build foo`' on VMS will not preserve the case of 'foo'. Rather than forcing people to write '`@Build "foo"`' we'll dispatch case-insensitively.

### `manpage_separator`

Use '`__`' instead of '`::`'.

### `prefixify`

Prefixify taking into account VMS' filepath syntax.

### `_quote_args`

Command-line arguments (but not the command itself) must be quoted to ensure case preservation.

### `have_forkpipe`

There is no native `fork()`, so some constructs depending on it are not available.

### `_backticks`

Override to ensure that we quote the arguments but not the command.

### `do_system`

Override to ensure that we quote the arguments but not the command.

### `_infer_xs_spec`

Inherit the standard version but tweak the library file name to be something Dynaloader can find.

### `rscan_dir`

Inherit the standard version but remove dots at end of name. This may not be necessary if `File::Find` has been fixed or `DECC$FILENAME_UNIX_REPORT` is in effect.

### `dist_dir`

Inherit the standard version but replace embedded dots with underscores because a dot is the directory delimiter on VMS.

### `man3page_name`

Inherit the standard version but chop the extra manpage delimiter off the front if there is one. The VMS version of `splitdir(['.foo'])` returns `"`, 'foo'.

### `expand_test_dir`

Inherit the standard version but relativize the paths as the native `glob()` doesn't do that for us.

### `_detildefy`

The home-grown `glob()` does not currently handle tildes, so provide limited support here.

Expect only UNIX format file specifications for now.

find\_perl\_interpreter

On VMS, `$^X` returns the fully qualified absolute path including version number. It's logically impossible to improve on it for getting the perl we're currently running, and attempting to manipulate it is usually lossy.

## AUTHOR

Michael G Schwern <schwern@pobox.com> Ken Williams <kwilliams@cpan.org> Craig A. Berry <craigberry@mac.com>

## SEE ALSO

perl(1), Module::Build(3), ExtUtils::MakeMaker(3)