NAME

File::Spec::VMS - methods for VMS file specs

SYNOPSIS

```
 require File::Spec::VMS; # Done internally by File::Spec if needed
```

DESCRIPTION

See File::Spec::Unix for a documentation of the methods provided there. This package overrides the implementation of these methods, not the semantics.

The mode of operation of these routines depend on the VMS features that are controlled by the DECC features `DECC$FILENAME_REPORT_UNIX` and `DECC$EFS_CHARSET`.

Perl needs to be at least at 5.10 for these feature settings to work. Use of them on older perl versions on VMS will result in unpredictable operations.

The default and traditional mode of these routines have been to expect VMS syntax on input and to return VMS syntax on output, even when Unix syntax was given on input.

The default and traditional mode is also incompatible with the VMS `EFS`, Extended File system character set, and with running Perl scripts under <GNV>, Gnu is not VMS, an optional Unix like runtime environment on VMS.

If the `DECC$EFS_CHARSET` feature is enabled, These routines will now accept either VMS or UNIX syntax. If the input parameters are clearly VMS syntax, the return value will be in VMS syntax. If the input parameters are clearly in Unix syntax, the output will be in Unix syntax.

This corresponds to the way that the VMS C library routines have always handled filenames, and what a programmer who has not specifically read this pod before would also expect.

If the `DECC$FILENAME_REPORT_UNIX` feature is enabled, then if the output syntax can not be determined from the input syntax, the output syntax will be UNIX. If the feature is not enabled, VMS output will be the default.

canonpath (override)

> Removes redundant portions of file specifications according to the syntax detected.

catdir (override)

> Concatenates a list of file specifications, and returns the result as a directory specification. No check is made for "impossible" cases (e.g. elements other than the first being absolute filespecs).

catfile (override)

> Concatenates a list of directory specifications with a filename specification to build a path.

curdir (override)

> Returns a string representation of the current directory: '[]' or '.'

devnull (override)

> Returns a string representation of the null device: '_NLA0:' or '/dev/null'

rootdir (override)

> Returns a string representation of the root directory: 'SYS$DISK:[000000]' or '/'

tmpdir (override)

> Returns a string representation of the first writable directory from the following list or '' if none are writable:

```
        /tmp if C<DECC$FILENAME_REPORT_UNIX> is enabled.
        sys$scratch:
        $ENV{TMPDIR}
```

Since perl 5.8.0, if running under taint mode, and if $ENV{TMPDIR} is tainted, it is not used.

updir (override)

Returns a string representation of the parent directory: '[-]' or '..'

case_tolerant (override)

VMS file specification syntax is case-tolerant.

path (override)

Translate logical name DCL$PATH as a searchlist, rather than trying to `split` string value of `$ENV{'PATH'}`.

file_name_is_absolute (override)

Checks for VMS directory spec as well as Unix separators.

splitpath (override)

```
    ($volume,$directories,$file) = File::Spec->splitpath( $path );
    ($volume,$directories,$file) = File::Spec->splitpath( $path,
$no_file );
```

Passing a true value for `$no_file` indicates that the path being split only contains directory components, even on systems where you can usually (when not supporting a foreign syntax) tell the difference between directories and files at a glance.

splitdir (override)

Split a directory specification into the components.

catpath (override)

Construct a complete filespec.

abs2rel (override)

Attempt to convert a file specification to a relative specification. On a system with volumes, like VMS, this may not be possible.

rel2abs (override)

Return an absolute file specification from a relative one.

COPYRIGHT

SEE ALSO

See *File::Spec* and *File::Spec::Unix*. This package overrides the implementation of these methods, not the semantics.

An explanation of VMS file specs can be found at *http://h71000.www7.hp.com/doc/731FINAL/4506/4506pro_014.html#apps_locating_naming_files*.