NAME

lib - manipulate @INC at compile time

SYNOPSIS

```
use lib LIST;


no lib LIST;
```

DESCRIPTION

This is a small simple module which simplifies the manipulation of `@INC` at compile time.

It is typically used to add extra directories to perl's search path so that later `use` or `require` statements will find modules which are not located on perl's default search path.

Adding directories to `@INC`

The parameters to `use lib` are added to the start of the perl search path. Saying

```
use lib LIST;
```

is *almost* the same as saying

```
BEGIN { unshift(@INC, LIST) }
```

For each directory in LIST (called `$dir` here) the lib module also checks to see if a directory called `$dir`/$archname/auto exists. If so the `$dir`/$archname directory is assumed to be a corresponding architecture specific directory and is added to `@INC` in front of `$dir`. lib.pm also checks if directories called `$dir`/$version and `$dir`/$version/$archname exist and adds these directories to `@INC`.

The current value of `$archname` can be found with this command:

```
perl -V:archname
```

The corresponding command to get the current value of `$version` is:

```
perl -V:version
```

To avoid memory leaks, all trailing duplicate entries in `@INC` are removed.

Deleting directories from `@INC`

You should normally only add directories to `@INC`. If you need to delete directories from `@INC` take care to only delete those which you added yourself or which you are certain are not needed by other modules in your script. Other modules may have added directories which they need for correct operation.

The `no lib` statement deletes all instances of each named directory from `@INC`.

For each directory in LIST (called `$dir` here) the lib module also checks to see if a directory called `$dir`/$archname/auto exists. If so the `$dir`/$archname directory is assumed to be a corresponding architecture specific directory and is also deleted from `@INC`.

Restoring original `@INC`

When the lib module is first loaded it records the current value of `@INC` in an array `@lib::ORIG_INC`. To restore `@INC` to that value you can say

```
@INC = @lib::ORIG_INC;
```

CAVEATS

In order to keep lib.pm small and simple, it only works with Unix filepaths. This doesn't mean it only works on Unix, but non-Unix users must first translate their file paths to Unix conventions.

```
# VMS users wanting to put [.stuff.moo] into
# their @INC would write
use lib 'stuff/moo';
```

NOTES

In the future, this module will likely use File::Spec for determining paths, as it does now for Mac OS (where Unix-style or Mac-style paths work, and Unix-style paths are converted properly to Mac-style paths before being added to `@INC`).

If you try to add a file to `@INC` as follows:

```
 use lib 'this_is_a_file.txt';
```

`lib` will warn about this. The sole exceptions are files with the `.par` extension which are intended to be used as libraries.

SEE ALSO

FindBin - optional module which deals with paths relative to the source file.

PAR - optional module which can treat `.par` files as Perl libraries.

AUTHOR

Tim Bunce, 2nd June 1995.

`lib` is maintained by the perl5-porters. Please direct any questions to the canonical mailing list. Anything that is applicable to the CPAN release can be sent to its maintainer, though.

Maintainer: The Perl5-Porters <perl5-porters@perl.org>

Maintainer of the CPAN release: Steffen Mueller <smueller@cpan.org>

COPYRIGHT AND LICENSE

This package has been part of the perl core since perl 5.001. It has been released separately to CPAN so older installations can benefit from bug fixes.

This package has the same copyright and license as the perl core.