

**NAME**

ExtUtils::MM\_Win32 - methods to override UN\*X behaviour in ExtUtils::MakeMaker

**SYNOPSIS**

```
use ExtUtils::MM_Win32; # Done internally by ExtUtils::MakeMaker if needed
```

**DESCRIPTION**

See [ExtUtils::MM\\_Unix](#) for a documentation of the methods provided there. This package overrides the implementation of these methods, not the semantics.

**Overridden methods****dlsyms**

replace\_manpage\_separator

Changes the path separator with .

**maybe\_command**

Since Windows has nothing as simple as an executable bit, we check the file extension.

The PATHEXT env variable will be used to get a list of extensions that might indicate a command, otherwise .com, .exe, .bat and .cmd will be used by default.

**init\_DIRFILESEP**

Using for Windows.

**init\_tools**

Override some of the slower, portable commands with Windows specific ones.

**init\_others**

Override the default link and compile tools.

LDLOADLIBS's default is changed to `$Config{libs}`.

Adjustments are made for Borland's quirks needing -L to come first.

**init\_platform**

Add MM\_Win32\_VERSION.

**platform\_constants****constants**

Add MAXLINELENGTH for dmake before all the constants are output.

**special\_targets**

Add .USESHELL target for dmake.

**static\_lib**

Changes how to run the linker.

The rest is duplicate code from MM\_Unix. Should move the linker code to its own method.

**dynamic\_lib**

Complicated stuff for Win32 that I don't understand. :(

**extra\_clean\_files**

Clean out some extra dll.{base,exp} files which might be generated by gcc. Otherwise, take out all \*.pdb files.

**init\_linker****perl\_script**

Checks for the perl program under several common perl extensions.

**xs\_o**

This target is stubbed out. Not sure why.

**pasthru**

All we send is -nologo to nmake to prevent it from printing its damned banner.

`arch_check` (override)

Normalize all arguments for consistency of comparison.

`oneliner`

These are based on what `command.com` does on Win98. They may be wrong for other Windows shells, I don't know.

`cd` `dmake` can handle Unix style `cd`'ing but `nmake` (at least 1.5) cannot. It wants:

```
cd dir1\dir2
command
another_command
cd ..\..
```

`max_exec_len`

`nmake` 1.50 limits command length to 2048 characters.

`os_flavor`

Windows is Win32.

`cflags`

Defines the `PERLDLL` symbol if we are configured for static building since all code destined for the `perl5xx.dll` must be compiled with the `PERLDLL` symbol defined.