

**NAME**

ExtUtils::Command - utilities to replace common UNIX commands in Makefiles etc.

**SYNOPSIS**

```
perl -MExtUtils::Command -e cat files... > destination
perl -MExtUtils::Command -e mv source... destination
perl -MExtUtils::Command -e cp source... destination
perl -MExtUtils::Command -e touch files...
perl -MExtUtils::Command -e rm_f files...
perl -MExtUtils::Command -e rm_rf directories...
perl -MExtUtils::Command -e mkpath directories...
perl -MExtUtils::Command -e eqtime source destination
perl -MExtUtils::Command -e test_f file
perl -MExtUtils::Command -e chmod mode files...
...
```

**DESCRIPTION**

The module is used to replace common UNIX commands. In all cases the functions work from @ARGV rather than taking arguments. This makes them easier to deal with in Makefiles.

```
perl -MExtUtils::Command -e some_command some files to work on
```

**NOT**

```
perl -MExtUtils::Command -e 'some_command qw(some files to work on)'
```

For that use *Shell::Command*.

Filenames with \* and ? will be glob expanded.

**cat**

```
cat file ...
```

Concatenates all files mentioned on command line to STDOUT.

**eqtime**

```
eqtime source destination
```

Sets modified time of destination to that of source.

**rm\_rf**

```
rm_rf files or directories ...
```

Removes files and directories - recursively (even if readonly)

**rm\_f**

```
rm_f file ...
```

Removes files (even if readonly)

**touch**

```
touch file ...
```

Makes files exist, with current timestamp

**mv**

```
mv source_file destination_file
mv source_file source_file destination_dir
```

Moves source to destination. Multiple sources are allowed if destination is an existing directory.

Returns true if all moves succeeded, false otherwise.

cp

```
cp source_file destination_file
cp source_file source_file destination_dir
```

Copies sources to the destination. Multiple sources are allowed if destination is an existing directory.

Returns true if all copies succeeded, false otherwise.

chmod

```
chmod mode files ...
```

Sets UNIX like permissions 'mode' on all the files. e.g. 0666

mkpath

```
mkpath directory ...
```

Creates directories, including any parent directories.

test\_f

```
test_f file
```

Tests if a file exists

dos2unix

```
dos2unix files or dirs ...
```

Converts DOS and OS/2 linefeeds to Unix style recursively.

## SEE ALSO

Shell::Command which is these same functions but take arguments normally.

## AUTHOR

Nick Ing-Simmons [ni-s@cpan.org](mailto:ni-s@cpan.org)

Currently maintained by Michael G Schwern [schwern@pobox.com](mailto:schwern@pobox.com).