

NAME

find2perl - translate find command lines to Perl code

SYNOPSIS

```
find2perl [paths] [predicates] | perl
```

DESCRIPTION

find2perl is a little translator to convert find command lines to equivalent Perl code. The resulting code is typically faster than running find itself.

“paths” are a set of paths where find2perl will start its searches and “predicates” are taken from the following list.

! PREDICATE

Negate the sense of the following predicate. The ! must be passed as a distinct argument, so it may need to be surrounded by whitespace and/or quoted from interpretation by the shell using a backslash (just as with using `find(1)`).

(PREDICATES)

Group the given PREDICATES. The parentheses must be passed as distinct arguments, so they may need to be surrounded by whitespace and/or quoted from interpretation by the shell using a backslash (just as with using `find(1)`).

PREDICATE1 PREDICATE2

True if `_both_` PREDICATE1 and PREDICATE2 are true; PREDICATE2 is not evaluated if PREDICATE1 is false.

PREDICATE1 -o PREDICATE2

True if either one of PREDICATE1 or PREDICATE2 is true; PREDICATE2 is not evaluated if PREDICATE1 is true.

-follow

Follow (dereference) symlinks. The checking of file attributes depends on the position of the `-follow` option. If it precedes the file check option, an `stat` is done which means the file check applies to the file the symbolic link is pointing to. If `-follow` option follows the file check option, this now applies to the symbolic link itself, i.e. an `lstat` is done.

-depth

Change directory traversal algorithm from breadth-first to depth-first.

-prune

Do not descend into the directory currently matched.

-xdev

Do not traverse mount points (prunes search at mount-point directories).

-name GLOB

File name matches specified GLOB wildcard pattern. GLOB may need to be quoted to avoid interpretation by the shell (just as with using `find(1)`).

-iname GLOB

Like `-name`, but the match is case insensitive.

-path GLOB

Path name matches specified GLOB wildcard pattern.

-ipath GLOB

Like `-path`, but the match is case insensitive.

-perm PERM

Low-order 9 bits of permission match octal value PERM.

-perm -PERM

The bits specified in PERM are all set in file's permissions.

- type X**
The file's type matches perl's **-X** operator.
- fstype TYPE**
Filesystem of current path is of type TYPE (only NFS/non-NFS distinction is implemented).
- user USER**
True if USER is owner of file.
- group GROUP**
True if file's group is GROUP.
- nouser**
True if file's owner is not in password database.
- nogroup**
True if file's group is not in group database.
- inum INUM**
True file's inode number is INUM.
- links N**
True if (hard) link count of file matches N (see below).
- size N**
True if file's size matches N (see below) N is normally counted in 512-byte blocks, but a suffix of "c" specifies that size should be counted in characters (bytes) and a suffix of "k" specifies that size should be counted in 1024-byte blocks.
- atime N**
True if last-access time of file matches N (measured in days) (see below).
- ctime N**
True if last-changed time of file's inode matches N (measured in days, see below).
- mtime N**
True if last-modified time of file matches N (measured in days, see below).
- newer FILE**
True if last-modified time of file matches N.
- print**
Print out path of file (always true). If none of **-exec**, **-ls**, **-print0**, or **-ok** is specified, then **-print** will be added implicitly.
- print0**
Like **-print**, but terminates with 0 instead of n.
- exec OPTIONS ;**
exec() the arguments in OPTIONS in a subprocess; any occurrence of {} in OPTIONS will first be substituted with the path of the current file. Note that the command "rm" has been special-cased to use perl's *unlink()* function instead (as an optimization). The ; must be passed as a distinct argument, so it may need to be surrounded by whitespace and/or quoted from interpretation by the shell using a backslash (just as with using [find\(1\)](#)).
- ok OPTIONS ;**
Like **-exec**, but first prompts user; if user's response does not begin with a y, skip the exec. The ; must be passed as a distinct argument, so it may need to be surrounded by whitespace and/or quoted from interpretation by the shell using a backslash (just as with using [find\(1\)](#)).
- eval EXPR**
Has the perl script *eval()* the EXPR.

`-ls`
Simulates `-exec ls -dils {} ;`

`-tar FILE`
Adds current output to tar-format FILE.

`-cpio FILE`
Adds current output to old-style cpio-format FILE.

`-ncpio FILE`
Adds current output to “new”-style cpio-format FILE.

Predicates which take a numeric argument N can come in three forms:

- * N is prefixed with a `+`: match values greater than N
- * N is prefixed with a `-`: match values less than N
- * N is not prefixed with either `+` or `-`: match only values equal to N

SEE ALSO

`find`, `File::Find`.