NAME

glob - global command (file name generation)

SYNOPSIS

glob *command* [*arg* ...]

DESCRIPTION

Glob is a port of the global command from Version 6 (V6) UNIX. It is passed an argument list by *tsh*(1) which contains one or more unquoted *pattern characters* (*****, **?**, **[**) and performs the actions described below on behalf of the shell.

Glob attempts to generate file-name arguments which match the given pattern arguments. The name of the specified *command* may also be given as a pattern. The meaning of each pattern character is as follows:

- The ***** character in a pattern matches any string of characters in a file name (including the null string).
- The **?** character in a pattern matches any single character in a file name.
- The **[...]** brackets in a pattern specifies a class of characters which matches any single file-name character in the class. Within the brackets, each character is taken to be a member of the class. A pair of characters separated by an unquoted - specifies the class as a range which matches each character lexically between the first and second member of the pair, inclusive. A - matches itself when quoted or when first or last in the class.

Any other character in a pattern matches itself in a file name.

Notice that the '.' character at the beginning of a file name, or immediately following a '/', is always special in that it must be matched explicitly. The same is true of the '/' character itself.

If the pattern contains no '/' characters, the current directory is always used. Otherwise, the specified directory is the one obtained by taking the pattern up to the last '/' before the first unquoted *****, **?**, or **[**. The matching process matches the remainder of the pattern after this '/' against the files in the specified directory.

If the argument contains no unquoted pattern characters, **glob** uses it as is. Otherwise, glob searches for file names in the current (or specified) directory which match the pattern. It then sorts them in ascending ASCII order, and the new sequence of arguments replaces the given pattern. The same process is carried out for each of the given arguments; the resulting lists are *not* merged. Finally, glob attempts to execute the command with the resulting argument list.

EXIT STATUS

If **glob** detects an error, it prints an appropriate diagnostic and exits with a non-zero status. Otherwise, the exit status is that of the executed command.

ENVIRONMENT

Notice that the concept of ‘user environment’ was not defined in Version 6 (V6) UNIX. Thus, use of the following environment variables by this port of the global command is an enhancement:

EXECSHELL

If set to a non-empty string, the value of this variable is taken as the path name of the shell which is invoked to execute the specified command when it does not begin with the proper magic number or a ‘#!shell’ sequence.

PATH

If set to a non-empty string, the value of this variable is taken as the sequence of directories which is used to search for the specified command. Notice that the global command from Version 6 (V6) UNIX always used the equivalent of ‘.:/bin:/usr/bin’, not PATH.

SEE ALSO

tsh(1)

Etsh home page: https://etsh.nl/

HISTORY

A **glob** command appeared as */etc/glob* in Version 1 (V1) UNIX.

AUTHORS

This port of the **glob** command is derived from Version 6 (V6) UNIX /usr/source/s1/glob.c. It was written by Ken Thompson of Bell Labs. Jeffrey Allen Neitzel <*jan@etsh.nl*> ported and maintains it as *glob*(1).

LICENSE

See either the LICENSE file which is distributed with **etsh** or https://etsh.nl/license/ for full details.