

## NAME

**plocate** – find files by name, quickly

## SYNOPSIS

**plocate** [*OPTION*]... *PATTERN*...

## DESCRIPTION

**plocate** finds all files on the system matching the given pattern (or all of the patterns if multiple are given). It does this by means of an index made by **updatedb**(8) or (less commonly) converted from another index by **plocate-build**(8).

**plocate** is largely argument-compatible with **mlocate**(1), but is significantly faster. In particular, it rarely needs to scan through its entire database, unless the pattern is very short (less than three bytes) or you want to search for a regular expression. It does not try to maintain compatibility with BSD **locate**, or non-UTF-8 filenames and locales. Most I/O is done asynchronously, but the results are synchronized so that output comes in the same order every time.

When multiple patterns are given, **plocate** will search for files that match *all* of them. This is the main incompatibility with **mlocate**(1), which searches for files that match one or more patterns, unless the **-A** option is given.

By default, patterns are taken to be substrings to search for. If at least one non-escaped globbing metacharacter (\*, ? or []) is given, that pattern is instead taken to be a glob pattern (which means it needs to start and end in \* for a substring match). If **--regexp** is given, patterns are instead taken to be (non-anchored) POSIX basic regular expressions, and if **--regex** is given, patterns are taken to be POSIX extended regular expressions. All of this matches **mlocate**(1) behavior.

Like **mlocate**(1), **plocate** shows all files visible to the calling user (by virtue of having read and execute permissions on all parent directories), and none that are not, by means of running with the setgid bit set to access the index (which is built as root), but by testing visibility as the calling user.

## OPTIONS

**-A, --all**

Ignored for compatibility with **mlocate**(1).

**-b, --basename**

Match only against the file name portion of the path name, ie., the directory names will be excluded from the match (but still printed). This does not speed up the search, but can suppress uninteresting matches.

**-c, --count**

Do not print each match. Instead, count them, and print out a total number at the end.

**-d, --database DBPATH**

Find matches in the given database, instead of **/var/lib/plocate/plocate.db**. This argument can be given multiple times, to search multiple databases. It is also possible to give multiple databases in one argument, separated by **:**. (Any character, including **:** and **\**, can be escaped by prepending a **\**.)

**-e, --existing**

Print only entries that refer to files existing at the time **locate** is run. Note that unlike **mlocate**(1), symlinks are not followed by default (and indeed, there is no option to change this).

**-i, --ignore-case**

Do a case-insensitive match as given by the current locale (default is case-sensitive, byte-by-byte match). Note that **plocate** does not support the full range of Unicode case folding rules; in particular, searching for *ß* will not give you matches on *ss* even in a German locale. Also note that this option will be somewhat slower than a case-sensitive match, since it needs to generate more candidates for searching the index.

**-p, --ignore-spaces**

Ignore punctuation and spaces when matching patterns.

**-l, --limit LIMIT**

Stop searching after *LIMIT* matches have been found. If **--count** is given, the number printed out will be at most *LIMIT*.

**-N, --literal**

Print entry names without quoting. Normally, **plocate** will escape special characters in filenames, so that they are safe for consumption by typical shells (similar to the GNU coreutils *shell-escape-always* quoting style), unless printing to a pipe, but this options will turn off such quoting.

**-0, --null**

Instead of writing a newline after every match, write a NUL (ASCII 0). This is useful for creating unambiguous output when it is to be processed by other tools (like **xargs(1)**), as filenames are allowed to contain embedded newlines.

**-r, --regexp**

Patterns are taken to be POSIX basic regular expressions. See **egex(7)** for more information. Note that this forces a linear scan through the entire database, which is slow.

**--regex**

Like **--regexp**, but patterns are instead taken to be POSIX *extended* regular expressions.

**-w, --wholename**

Match against the entire path name. This is the default, so unless **-b** is given first (see above), it will not do anything. This option thus exists only as compatibility with **mlocate(1)**.

**--help** Print out usage information, then exit successfully.

**--version**

Print out version information, then exit successfully.

**ENVIRONMENT****LOCATE\_PATH**

If given, appended after the list of **--database** paths (whether an explicit is given or the default is used). Colon-delimiting and character escaping follows the same rules as for **--database**.

**AUTHOR**

Steinar H. Gunderson <steinar+plocate@gunderson.no>

**SEE ALSO****plocate-build(8), mlocate(1), updatedb(8)**