dust Visual disk usage

Walk dust <dir> → `-d` / `-x` / `-X` → **Filter** depth, size, regex → `-n` / `-r` → **Display** tree with bars → `-j` → **Pipe** JSON to jq

CONCEPTS

dust is a du with a tree and bars: the largest entries come first, each with its share of the total.

Options can be set permanently in ~/.config/dust/config.toml, using the long flag names.

SCOPE

`dust <dir> <dir>`
Analyse several directories

`dust -d <n>`
Show only n levels deep

`dust -n <n>`
Show n entries

`dust -D`
Directories only

`dust -F`
Files only: largest files

SCOPE (cont.)

`dust -x`
Stay on one filesystem

`dust -L`
Follow symlinks

`dust -i`
Hide hidden files

`dust -f`
Count files instead of size

FILTER

`dust -X <name>`
Ignore files or dirs by name

`dust -e <regex>`
Only entries matching regex

`dust -v <regex>`
Exclude entries matching regex

`dust -z <size>`
Hide entries below a size

FILTER (cont.)

`dust -t`
Group by file type

`dust --collapse <name>`
Keep a folder collapsed

DISPLAY

`dust -r`
Reverse order: biggest at top

`dust -p`
Show full paths

`dust -s`
Apparent size, not disk use

`dust -o <fmt>`
Size unit: si, b, kb, kib, mb, gb, etc

`dust -b`
Hide bars and percentages

`dust -B`
Bars on the right

DISPLAY (cont.)

`dust -c`
No colours

`dust -C`
Force colours

`dust --dim`
Dim the bars

`dust --skip-total`
Omit the total row

`dust -P`
No progress indicator

`dust -R`
Screen-reader friendly output

INPUT AND OUTPUT

`dust -j`
Print JSON

`dust --files-from <file>`
Read paths, newline separated

INPUT AND OUTPUT (cont.)

`dust --files0-from <file>`
Read paths, NUL separated

`dust -T <n>`
Walker threads, useful on NFS

RECIPES

What fills this disk

```
dust -x -d 2 /
```

Biggest files in a project

```
dust -F -n 20 <dir>
```

Skip noise, find big images

```
dust -X node_modules -X .git -e '\.(png|jpg)$' <dir>
```

Feed paths from find

```
find <dir> -name '*.log' -print0 | dust --files0-from -
```