gdu Fast disk usage

START HERE

Up / **Down**
Move the selection

Enter / **Right**
Open the directory

Left / **h**
Back to the parent

s
Sort by size

d
Delete the selection

/
Search by name

?
Show help

q
Quit gdu

Scan gdu <path> `-o` / `-f` → **Browse** sort, filter, info `d` / `D` / `e` → **Clean up** delete, trash, empty

CONCEPTS

Start gdu in the directory you suspect, then dive into the biggest entry and repeat; the bar shows each entry's share.

Scans use many cores, great on SSDs; add --sequential on a spinning disk.

Hard links are counted once. Default sizes are disk usage; press a to compare with apparent size.

NAVIGATE

Up / **Down**
Move the selection

k / **j**
Move up / down (vi keys)

Enter / **Right**
Open the selected directory

l Open the directory (vi key)

Left / **h**
Go to the parent directory

PgUp / **PgDn**
Move a page at a time

g / **G**
Jump to the top / bottom

/ Search entries by name

? Show the help window

NAVIGATE (cont.)

q
Quit (asks after a long scan)

Q
Quit and print the current directory
Handy for a shell wrapper that cds to where you ended up.

SORT AND VIEW

s
Sort by size; again to reverse

n
Sort by name; again to reverse

C Sort by item count

M Sort by mtime

a
Toggle disk usage / apparent size

c Show or hide item counts

m Show or hide latest mtime

B
Align bars to the biggest entry

T
Filter by file type (extension)

ACT ON ITEMS

d Delete the selected item
Asks for confirmation. Disabled by --no-delete.

ACT ON ITEMS (cont.)

D Move the item to the trash
Uses the built-in trash unless --trash-command is set.

e Empty the directory or file

i Show info about the item

v View the file's content

r
Rescan the current directory

Space
Mark an item for deletion

y
Copy the item's path to the clipboard

o
Open in an external program

I Ignore the item

p Print marked paths on exit

b
Spawn a shell in this directory

E
Export the analysis as JSON

Mark several items with Space, then delete them in one go.

DURING A SCAN

Tab
Preview results found so far

DURING A SCAN (cont.)

Esc
Stop scanning, keep results

Ctrl + **C**
Stop scanning; again to quit
With --ctrl-c-quits the first press quits.

RUN IT

gdu
Analyse the current directory

gdu *<path>*
Analyse a directory

gdu *<dir1>* *<dir2>*
Analyse several directories

gdu -d
List mounted disks

gdu -x *<path>*
Stay on one filesystem

gdu -a
Start on apparent size

gdu -C
Show item counts

gdu -M
Show latest mtime

gdu --si
Use kB / MB, not KiB / MiB

gdu --collapse-path
Merge single-child directory chains

gdu --web *<path>*
Browse in a web UI

FILTER THE SCAN

gdu -i *<dir>*,*<dir>* *<path>*
Ignore paths (default /proc,/dev,/sys,/run)

gdu -I '*<regex>***'** *<path>*
Ignore paths by pattern

gdu -X *<file>* *<path>*
Read ignore patterns from a file

gdu -G .gitignore *<path>*
Ignore using gitignore-style rules

gdu -H *<path>*
Skip hidden directories

gdu -T *<ext>*,*<ext>* *<path>*
Include only these file types

gdu --min-age 30d *<path>*
Only files at least 30 days old

NON-INTERACTIVE AND EXPORT

gdu -n *<path>*
Print sizes, no TUI

gdu -ps *<path>*
Print only the total

gdu -t 10 *<path>*
Show the 10 largest files

gdu -n --depth 2 *<path>*
Print the tree to depth 2

gdu -o- *<path>* **| gzip > scan.json.gz**
Export a scan as JSON

NON-INTERACTIVE AND EXPORT (cont.)

zcat scan.json.gz | gdu -f-
Browse a saved scan

SAFETY AND CONFIG

gdu --no-delete
Disable all deletion

gdu --trash-command '*<command>***'**
Own command for the D key
Example: 'trash-put --trash-dir ~/mytrash'.

gdu -m *<n>*
Limit CPU cores used

RECIPES

Find what fills the disk

```
gdu -x /
# s sorts by size, Enter dives in,
# d or D cleans up
```

Quick check, no UI

```
gdu -t 10 ~/projects
gdu -ps ~/Downloads
```

Scan big, browse later

```
gdu -o- / | gzip -c > report.json.gz
zcat report.json.gz | gdu -f-
```

Review safely, skip build output

```
gdu --no-delete -G .gitignore ~/projects
gdu -H ~/projects
```