df and du Disk space usage

df disk is full? → which mount → **du -x** which directory? → drill down → **du -d 1** which subdir / file?

CONCEPTS

df asks the filesystem how full it is; du walks the tree and adds up files. Their numbers can differ.

These are GNU flags. BSD/macOS df and du differ: no --inodes, --apparent-size or --output.

Full disk but df -i shows 100% IUse? You ran out of inodes (many tiny files), not bytes.

df and du disagree when a deleted file is still held open by a process; close or restart it to free the space (find it: lsof +L1).

DF: FREE SPACE

df -h
Human-readable sizes (powers of 1024)

df -h *<path>*
Only the filesystem containing <path>

df -hT
Show filesystem type column

DF: FREE SPACE (cont.)

df -i
Inodes instead of blocks

df -H
Sizes in powers of 1000 (like disk vendors)

df -t *<type>*
Only filesystems of this type

df -x tmpfs -x devtmpfs
Hide filesystem types

df -l
Local filesystems only (skip NFS)

df --total
Add a grand total row

df --output=source,avail,target
Pick the columns (see detail for fields)

df -a
Include pseudo and inaccessible filesystems

df -BG
Fixed unit, here gigabytes (1024^3)

df -P
POSIX format, one line per filesystem

df --sync
Sync before reading usage

DF: FREE SPACE (cont.)

df -k
1K blocks

DU: DIRECTORY SIZE

du -sh *<dir>*
Total for one directory

du -h -d 1 *<dir>*
Size of each direct subdirectory

du -sh *
Size of each item in the cwd

du -ah *<dir>*
Every file, not just directories

du -h -d 1 *<dir>* **| sort -h**
Sort smallest to biggest

du -ah *<dir>* **| sort -rh | head**
Top 10 space users

du -xh -d 1 *<dir>*
Don't cross filesystem boundaries

du -ch *<file>...*
Per item plus grand total

du -t *<size>* **-ah** *<dir>*
Skip entries under <size> (negative: over)

DU: DIRECTORY SIZE (cont.)

du --exclude='*<pattern>***'** *<dir>*
Skip matching names

du --apparent-size -sh *<dir>*
Content size, not disk blocks

du -S *<dir>*
Exclude subdirectories from each dir's size

du --time -h -d 1 *<dir>*
Add newest modification time

du --inodes -d 1 *<dir>* **| sort -n**
File count per directory

du -L *<dir>*
Follow all symlinks

du -l *<dir>*
Count hard-linked files each time

du -b *<file>*
Exact bytes (apparent size)

du -m *<dir>*
Sizes in MiB (1024^2)

du --si -s *<dir>*
Powers of 1000

du -X *<file>*
Read exclude patterns from a file

DU: DIRECTORY SIZE (cont.)

du -0 *<dir>* **| sort -z**
NUL-terminated lines for odd filenames

du --files0-from=*<file>*
Read NUL-separated names from a file or -

du --time=ctime --time-style=long-iso
Choose the time kind and format

RECIPES

Find what fills the disk

```
df -h
# pick the full mount, then descend
du -xh -d 1 / | sort -h | tail
du -xh -d 1 /var | sort -h | tail
```

Biggest files under a tree

```
du -ah /var 2>/dev/null | sort -rh | head -n 20
```

Size of files chosen by find

```
find . -name '*.log' -print0 | du --files0-from=- -ch | tail -n 1
```

Alert when a disk is nearly full

```
df --output=pcent,target | tail -n +2 | awk '$1+0 > 90 {print "low space:", $2}'
```