

NAME

ls — list file properties and directory contents

SYNOPSIS

```
ls [-iszk] [-p | --file-type | -F] [--colo[u]r] [--hyperlink]
  [-A | -a] [-B] [-I ignore-pattern] --hide=pattern...
  [-d | -R] [-H | -L] --dereference-command-line-symlink-to-dir
  [-l | -x | -C | -m] [-l | -o | -g | -n [-G] [--author]] [-D] [-w output-width]
  [-T tab-width]
  [-r] [-t | -S | -X | -v | -U | -f | --sort=width] [--group-directories-first]
  [-u | -c | --time=birth]
  [-, | -h | --si | --block-size=block-size]
  [-N | -b | -Q] --quoting-style=literal|shell{,-escape}{,-always}
|escapec{,-maybe}{[c]locale} [-q] [--zero]
  [--time-style=[posix-]{long|full}-iso
  |+date-format-old [HL date-format-recent]]
  [file]...
```

DESCRIPTION

Writes the names and optionally the file type (**-p**, **--file-type**, **-F**), the i-node numbers (**-i**), space used (**-s**), and the file mode, link count, ownership, size, modification time, symbolic link target (**-l**) of files found under *files* (or ".", the default) to the standard output stream.

The filenames are laid out in a table, escaped for pasting into the shell (if it is a teletype), or one per line, with unadulterated filenames (otherwise). **-R** allows recursion and examining entire directory hierarchies.

By default, if a *file* is a symbolic link to a directory, its contents are thus listed. With **-Fl****d**, the symbolic link itself is examined. Otherwise, symbolic links are not followed by default, and **-HD** govern them.

Each *file* which is not a directory is examined directly. If **-d**, this applies to all *files*; otherwise, *files* which are directories are read, and their contents are listed (under a heading, if required). Not all directory entries are shown by default: filenames starting with a dot "." are hidden. They may be shown by using **-Aa**. Additional filtering is available with **-BI**, **--hide**.

All filenames shown are sorted:

by default,	by their contents, according to the collation order of the current locale,
with -S	by file size, smallest file first,
with -t	by the modification time (but see below), most recently modified first,
with -X	by the bit after the last dot (.) in the filename,
with -v	by their contents interpreted as a version,
with --sort=width	by their displayed width, after any quoting and escaping,
with -S	not at all.

All orderings (except **-vS**) are disambiguated by the default filename sort.

-r reverses the sort, and **--group-directories-first** can be used to additionally partition all (symbolic links to) directories to the front.

The time used for sorting and shown in **-l** mode is

by default,	the modification time (<i>st_mtim</i>),
with -u	the access time (<i>st_atim</i>),
with -c	the i-node status change time (<i>st_ctim</i>),
with --time=birth	the i-node birth time (<i>stx_btime</i> (Linux), <i>st_birthtim</i> (NetBSD, FreeBSD)), else the i-node status change time (-c , <i>st_ctim</i>).

-p, **--file-type**, **-F** decorate filenames with a single character that indicates their type:

*	/	@		=	%	>
regular file, executable by anyone	directory	symbolic link	FIFO	socket	white-out	door

Output formats

If the standard output stream is a teletype, the output for each directory is organised into a table, with consecutive files going downward (**-C**). Otherwise, one file is listed per line (**-1**).

The following examples show the layout of files [1, 10], on a 20-column output:

-1: 1	-x (table, rows filled):
2	1 2 3 4 5 6 7
3	8 9 10
4	-C (table, columns filled):
5	1 3 5 7 9
6	2 4 6 8 10
7	-m (commas, no alignment):
8	1, 2, 3, 4, 5, 6, 7,
9	8, 9, 10
10	

With **-isZ**, additional information is tacked on before each filename: the i-node number (*st_ino*), the space used by the file, in kibibytes (or in a human-readable size with **-h**), rounded up (*st_blocks*), the MAC context, if any, else "?". These are added in this order. **-s** (and **-1**) also add a line reading "total *blocks*" before each listed directory: this is the *sum* of the occupied space (*st_blocks*) for each *shown* file, in 512-byte increments by default (**-k** and **-,** show kibibytes; **-h** affects this too). If a file appears in a directory many times, it *will* be counted many times. This is in contrast to `du(1)`.

Observe how these affect our layout (file *a* is i-node 41 and takes up 1KiB; *b* — 303 and takes up 20KiB; *c* — 502 and takes up 400KiB):

-1: 41 1 a	-x: 41 1 a 303 20 b
303 20 b	502 400 c
502 400 c	
-C: 41 1 a 502 400 c	-m: 41 1 a, 303 200 b,
303 200 b	502 400 c

-long output format

```

-rw-r--r-- 1 cicada users      768 2022-08-01 1
-rw-r--r-- 2 cicada users  16248 07-01 16:06 2
-rw-r--r--+ 1 cicada users 1024000 06-29 01:35 3
↑file mode↑ ↑owner  group size/  time      filename
|           | link count  device
type      ACL/MAC

```

and these are

<i>type</i>	One of: - d l p s b c w indicating: regular file directory symbolic link FIFO socket block device character device white-out or ? if unknown. (<i>st_mode</i>)
<i>file mode</i>	the file access mode (reading, writing, execution, and special bits, <i>st_mode</i>); see <code>chmod(1)</code> for more on this.
<i>ACL/MAC</i>	"+" if the file has an associated <code>acl(5)</code> , "." if the file has a MAC label, otherwise empty. "+" overrides ".".
<i>link count</i>	<i>st_nlink</i> : for regular files, this indicates how many times this file (i-node) exists in the filesystem. On some filesystems, this may indicate the number of entries for directories. Otherwise tends to be meaningless.
<i>owner, group</i>	The user and group that own this file, together with the <i>file mode</i> this dictates who can do what with it (<i>st_uid</i> , <i>st_gid</i>). These are looked up in <code>passwd(5)</code> , -n shows numbers instead. -g hides the owner and -o hides the group.
<i>size</i>	The "apparent" size: how big the file is. For symbolic links, this is the size of the symbolic link itself. On some filesystems, this may indicate the number of entries for directories. This is presented in bytes; -, separates digit groups (usually into thousands), -h formats this for human consumption, and --block-size can change the base unit (these also affect -s). (<i>st_size</i>)

device If the file being listed is of a device *type*, then the *size* column is replaced with the device number (*st_rdev*), formatted as decimal "*major, minor*".

time By default the modification time, but see above (**-u** &c.) for a way to select another time. This is formatted according to **--time-style** (*\$TIME_STYLE*). Two formats are available: for files younger than six months ("07-01 16:06") and older than six months (or from the future) ("2022-08-01 ").

For more on these fields see *inode(7)*.

-l is also affected by **-isZ**, as well as **--author**:

```
502 400 -rw-r--r--+ 1 cicada users cicada system_u:object_r:user_home_t:s0 1024
-i -s --author -Z
```

The author (*st_author*) is available on the Hurd only. The owner UID is used elsewhere.

In this mode, if a file is a symbolic link (and symbolic links aren't being followed (not **-HL**)), then the *filename* may be listed as: "*filename -> target*". Where *target* is the content of the symbolic link. In this case, the type decoration is skipped from the *filename*, but (if available) will appear on *target*.

Filename quoting

If the standard output stream is a teletype, each filename is quoted in such a way as to make it a single quoted word for a *sh(1)*-compatible shell (**--quoting-style=shell-escape**). Otherwise, filenames are output verbatim (**-N**). Beware of filenames containing newlines (and consider **--zero**).

Some quoting modes (**literal**, **shell{-always}**) may *not* quote control characters and other non-printables. If the standard output stream is a teletype, each is replaced with a "?" (**-q**).

The following **--quoting-styles** are available:

literal	-N	none; nonprintables subject to -q
shell		'-quoted if it contains whitespace or shell special characters (any of "\$&(){}*?;<=>[^\`)", or starts with a ~; 's escaped as \'; nonprintables subject to -q
shell-always		shell but always quote
shell-escape		shell but nonprintables are escaped with the <i>\$'\t'</i> <i>\$'\321\210'</i> syntax
shell-escape-always		shell-escape but always quote
escape	-b	space escaped as \ ; backslash escaped as \\\; nonprintables escaped as \t \321\210
c	-Q	"-quoted; closing quote escaped with a \; backslash escaped as \\\; nonprintables escaped as \t \321\210
c-maybe		c but only add quotes if any escaping happened
locale		c but use proper quotes appropriate for the locale
locale		locale but default to using ' quotes in the C (POSIX) locale

shell* may instead quote with a " if a ' was encountered and no non-printables were escaped and no shell specials were found.

With **-h** or **--si** sizes are output in a human-readable 3.2T-style. Otherwise, sizes are output in rounded-up blocks of **--block-size**, the first valid of the *LS_BLOCK_SIZE*, *BLOCK_SIZE*, *BLOCKSIZE* environment variables, or **1** (file sizes), **1024** (file blocks used), **512** (directory block total, unless **-k**), bytes.

-B, and the block size environment variables are in the case-insensitive format:

[*base*][**KMGTPPEZY**][**B**] (with at least one of {*base*, **KMGTPPEZY**, **B**})

Where *base* is an optionally-floating-point number of bytes, defaulting to **1**, which is then optionally multiplied by the relevant unit. **B** sets the unit multiplier to **1000** (from **1024**). The block size is equal to *base-unit^{mult}*, if any, or *base*.

With **-l**, the time format is the `date(1)`-style one specified by **--time-style**, the `TIME_STYLE` environment variable, or `"%b %e %H:%M"` for recent files and `"%b %e %Y"` for old ones. If a time is unrepresentable in the current time-zone, it's written as-if via `"%s.%N"` (*seconds.nanoseconds*) and a diagnostic is issued to the standard error stream.

OPTIONS

-i, --inode	Prepend each file with its i-node number.
-s, --size	Prepend each file with its space-used "size" (in kilobytes, subject to -l , -h , --si , --block-size).
-Z, --context	Prepend each file with its MAC context (or "?").
-k, --kibibytes	Format per-directory space-used totals as kilobytes instead of 512-byte blocks. Implied by -l .
-p	Decorate filenames with a / if they're directories.
-F, --classify	Decorate filenames by appending one of <code>*/@ =></code> depending on the file type (see above).
--file-type	-F but don't care about executable regular files ("*").
--indicator-style=none	Don't decorate filenames. This is the default.
--indicator-style=slash	-p
--indicator-style=file-type	--file-type
--indicator-style=classify	-F
--classify=always yes force	-F
--classify=auto tty if-tty	If the standard output stream is a teletype: -F . Otherwise: --indicator-style=none .
--classify=never none	--indicator-style=none
--colour, --color, --colour=always yes force, --color=always yes force	Colourise output according to <code>\$LS_COLORS</code> .
--colo[u]r=auto tty if-tty	If the standard output stream is a teletype: --colour . Otherwise: --colour=no .
--colo[u]r=never none	Don't colourise output. This is the default.
--hyperlink, --hyperlink=always yes force	Prepend each filename written with <code>"\e]8;;file://hostname/absolute-path-to-file\a"</code> , where <i>absolute-path-to-file</i> is %-escaped, and suffix it with <code>"\e]8;;\a"</code> .
--hyperlink=auto tty if-tty	If the standard output stream is a teletype: --hyperlink . Otherwise: --hyperlink=none .
--hyperlink=never none	Don't wrap filenames thus. This is the default.
-A, --almost-all	By default, filenames starting with a dot (".") are hidden. This shows all of them except "." and "..".
-a, --all	This shows all of them.
-I, --ignore=ignore-pattern	If a filename read from a directory matches <code>glob(7)</code> <i>ignore-pattern</i> , discard it.
--hide=pattern	The same, but don't honour <i>pattern</i> if -Aa was given.
-B, --ignore-backups	-I *~ -I .*~
-d, --directory	Don't open <i>files</i> which are directories, list them normally. Overrides -R .
-R, --recursive	For all children of <i>files</i> which are directories and which are not "." or "..": open them and list their contents as well. Repeat until a directory with no constituent directories or a loop is reached.
-H, --dereference-command-line:	Follow <i>files</i> which are symbolic links, but not any of their children.

- l, --dereference** Follow all symbolic links.
- dereference-command-line-symlink-to-dir**: Follow symbolic-link *file* if its target is a directory. This is the default unless **-Fld**.
- l, --format=single-column** Write each filename on its own line. **-l** is ignored if **-l** was also given (but **--format** isn't). This is the default if the standard output stream is not a teletype.
- x, --format=horizontal|across** Arrange filenames into a table as wide as the output allows, consecutive filenames fill rows first.
- C, --format=vertical** Arrange filenames into a table as wide as the output allows, consecutive filenames fill columns first. This is the default if the standard output stream is a teletype.
- m, --format=commas** Follow each filename by a comma, break before the output width.
- l, --format=long|verbose** Write the **-long output format**.
- o** **-l** but hide the group name.
- g** **-l** but hide the owner name.
- n, --numeric-uid-gid** **-l** but write owner and group as numbers directly instead of as names.
- G, --no-group** If **-l**, hide the group name.
- author** Add the author name after the group name on the Hurd. Repeat the owner name after the group name elsewhere.
- D, --dired** Append a line with `"/DIRED// start one-past-the-end..."` pairs of byte offsets of filenames listed in the output, if any, then a similar `"/SUBDIRED/"`-started line with offsets of listed directories, then `"/DIRED-OPTIONS// --quoting-style=..."`, duplicating **ls**'s; the offsets include all quoting; this output format is understood by emacs(1)'s Dired mode.
- w, --width=output-width** Lay out to fit *output-width*. **0** removes the line width limit and turns **-C** into **-x**. Defaults to `$COLUMNS` (if set and non-null), then the width of the teletype attached to the standard output, input, or error stream, then **80**.
- T, --tabsize=tab-width** When moving to the start of a table cell and *tab-width* > 1 use tabs, assumed to be laid out every *tab-width*th column, instead of spaces. Defaults to `$TABWIDTH` or **8**.
- r, --reverse** Reverse the sort order.
- t, --sort=time** Sort by the modification time (unless **-cu**), newest first. Disambiguate with the default sort order (filename by collation sequence).
- S, --sort=size** Sort by file size, smallest first. Disambiguate with the default sort order.
- X, --sort=extension** Sort by the bit of the filename after the last dot ("."). Disambiguate with the default sort order.
- v, --sort=version** Sort by filename collation sequence, modified to sort versions instead; this is the same as `sort(1) -V, q.v.`
- sort=width** Sort by column width of quoted filename. Disambiguate with the default sort order.
- U, --sort=none** Don't sort at all, output in the same order files were read from the directory.
- f** **-a** and overrides all sorting flags to **-U**
- group-directories-first** Unless **-U**, move all directories (and symbolic links to directories) to the front after sorting.

-u, --time=atime|access|use Sort by last access time. Write the last access time in the **-long output format**.

-c, --time=ctime|status Sort by i-node status change time. Write the i-node status change time in the **-long output format**.

--time=birth|creation Sort by i-node birth time on Linux, NetBSD, and FreeBSD and by i-node status change time elsewhere. Write the the same in the **-long output format**.

-, Format sizes (real and blocks-used) with thousands-separators (where the locale places them, so not necessarily every thousand). For example, this may turn a size of **4294967296** (4 gibibytes) into **4,294,967,296**. Implies **-k**. Overrides **-h**, **--si**, **--block-size**.

-h, --human-readable Format sizes (real and blocks-used) in a human readable *1024*-based *3.2T* style. Overrides **-,**, **--si**, **--block-size**.

--si Likewise, but *1000*. Overrides **-,**, **-h**, **--block-size**.

--block-size=block-size Set block size for size output. Overrides **-,**, **-h**, **--si**.

-N|-b|-Q|--quoting-style=literal|shell|{-escape|{-always|escape|c|{-maybe|}[c]locale
See **Filename quoting**.

--literal **-N**

--escape **-b**

--quote-name **-Q**

-q, --hide-control-chars Replace non-printable characters in the quoted filename with **?s**. This is the default if the standard output stream is a teletype.

--show-control-chars Don't. This is the default if the standard output stream is not a teletype.

--zero End lines with a NUL byte instead of a newline. Also **-N** **--colour=no**.

In the following, **H_L** indicates an *actual* newline.

--long-iso **-l --time-style=full-iso**

--time-style=iso Equivalent to **--time-style=+"%010F H_L%m-%d %R"** (*YYYY-MM-DD* — the ISO 8601 date format for old files and *MM-DD HH:MM* for recent files).

--time-style=long-iso Equivalent to **--time-style=+"%010F %R"** (*YYYY-MM-DD HH:MM*) for both file age groups.

--time-style=full-iso Equivalent to **--time-style=+"%010F %T.%N %z"** (*YYYY-MM-DD HH:MM:SS.NSNSNSNSN ±TZTZ*) for both file age groups.

--time-style=+date-format-old[H_Ldate-format-recent]
Format via *date(1)*-compatible *date-format-old* format. If the style contains a newline, use the bit before the newline for old files and the bit after the newline for recent ones.

--time-style=posix-iso|posix-long-iso|posix-full-iso|posix-+...
If in the **C** (**POSIX**) locale, use the default time style of **"%b %e %Y"** (*Mmm DD YYYY*) for old files and **"%b %e %H:%M"** (*Mmm DD HH:MM*) for recent ones. Otherwise, ignore the **posix-** prefix.

All **--indicator-style**, **--classify**, **--colo[u]r**, **--hyperlink**, **--format**, **--sort**, **--time**, **--quoting-style**, and non-+ **--time-style** values are prefix-matched (**--time=c** **--time-style=posix-f** is equivalent to **--time=ctime** **--time-style=posix-full-iso**, &c.).

ENVIRONMENT

LS_BLOCK_SIZE, BLOCK_SIZE, BLOCKSIZE

If any of these is valid, then the default becomes **--block-size** with the value of the first valid one.

COLUMNS

Override *output-width* for columnated **Output formats**.

TABWIDTH

Override the default **8** *tab-width*.

QUOTING_STYLE

Change the default **--quoting-style** from **shell-escape** or **-N**.

TIME_STYLE

Used as the time format if no **--time-style** instead of the default.

TZ

Override timezone for formatting file times, cf. `tzset(3)`.

LS_COLORS

If **--colo[u]r** was specified, parse this as a colon (":")-separated list of *context=colour* pairs, where *context* is any of

lc escape sequence, else "*escape* ["
rc end-escape sequence, else "m"
rs reset
ec reset sequence, else "{**lc**}{**rs**}{**rc**}"

su regular file u+s
sg regular file g+s
ex regular file executable by anyone
mh regular file *st_nlink*>1
pattern if filename matches case-insensitive glob *pattern*
fi regular file default

tw directory o+wt
st directory o+t
ow directory o+w
di directory default

ln symbolic link default
or symbolic link dangling
mi symbolic link's target if dangling; else use the real values

pi fifo
so socket
do door
bd blockdev
cd chardev

no otherwise-uncoloured text (file metadata)

and (except in the case of **lc**, **rc**, **rs** which are used verbatim) each *colour* is wrapped in **lc** and **rc** on either side, then written before the filename, and the filename is followed by **ec**. If **no** is set, the same applies to any pre-filename data. For example, if "**so**=01;35:*.mp3=00;36", then sockets will be coloured bold pink and MP3s teal.

These mostly don't interact, but: if **mh** isn't given (or it's a string of **0**s), the branch is skipped and multi-link files are coloured **fi**. The same applies to **mi** and **or**: if **mi** is empty or all **0**s, **or** is used. If **no** is set but **fi** isn't, **fi** copies **no**.

EXIT STATUS

2 if couldn't access or open one of the *files*

1 if couldn't access or open a directory or couldn't read a symbolic link,

SEE ALSO

date(1), dircolors(1), du(1), find(1), fstatat(2), statx(2), fnmatch(3), glob(7),
inode(7)

STANDARDS

Conforms to IEEE Std 1003.1-2024 ("POSIX.1") — only **-iskpFAa1xCmlognrtSfucdRHLq** are standard. BLOCKSIZE is an extension, originating from 4.4BSD. **-Z**, **--file-type**, **--color**, **--hyperlink**, **-BI**, **--hide**, **--dereference-command-line-symlink-to-dir**, **-G**, **--author**, **-DwTXvU**, **--sort**, **--group-directories-first**, **--time**, **-h**, **--si**, **--block-size**, **-NbQ**, **--quoting-style**, **-c**, **--maybe**, **--zero**, **--time-style** and LS_BLOCK_SIZE, BLOCK_SIZE TABWIDTH QUOTING_STYLE TIME_STYLE LS_COLORS are extensions, originating from the GNU system (but **-D** indents the output and only works with **-l** (and even then only sometimes) there). **-r** is an extension, originating from FreeBSD. The **--colour** spelling is an extension. FreeBSD **ls** also has **-Z** but it only works in the **-long output format**, puts it after the usernames, and **"-"** instead of **"?"**.

The block size for each file's **-s** blocks-used size is unspecified by the standard, and this implementation defaults to 1KiB. The block size for the directory total must be 512 unless **-k**. The GNU system brokenly defaults to **-k** unless the POSIXLY_CORRECT environment variable is set.

The GNU system's LS_COLORS also understands **cl**, **ca**; it's unclear what they do.

Devices are listed as **"0xdevice"** on FreeBSD and as **"major, 0xminor"** with 8-digit *minor* on DragonFly.

The standard requires \$COLUMNS takes precedent over the window size. The standard "expects" implementations to be able to "traverse arbitrary depths when processing the **-R** option". These are not the case on the GNU system. The latter is also not the case on the BSD with **-L**.

HISTORY**Research UNIX**

Appears in the UNIX Programmer's Manual as **ls(I)**:

NAME **ls** -- list contents of directory

SYNOPSIS **ls** [**-ltasd**] name₁ ...

with most of the output formatting semantics fully-formed. **-s** doesn't affect **-l** (because it's derived from rounding up the size and accounting for how additional extent i-nodes), **-l** behaves as-if **-i** (and the system doesn't have groups, and all times are expressed without a year anyway), and the default output mode is as-if **-l**. **-t** only compares the times, and doesn't disambiguate by names. Despite appearances, the default is **"."**. Thus,

```
# ls
a.out
core
ex
sdate.s
```

```
# ls -s
total    37
  1 a.out
 34 core
  1 ex
  1 sdate.s
```

```
# ls -l
total    37
211 s-rw-rw  1 root      36 Jan   1 00:00:00 a.out
213 l-rw-rw  1 root    16448 Jan   1 00:00:00 core
212 s-rw-rw  1 root      59 Jan   1 00:00:00 ex
214 s-rw-rw  1 root      31 Jan   1 00:00:00 sdate.s
```



```
# ls -lat
total    74
209 sdrwrw  2 root      90 Jan  1 00:00:00 .
 41 sdrwr- 10 root     110 Jan  1 00:00:00 ..
213 l-rwrw  2 root   16448 Jan  1 00:00:00 core
212 s-rwrw  1 root      59 Jan  1 00:00:00 ex
211 sxrwrw  1 123     128 Jan  1 00:00:00 a.out
214 s-rwrw  1 root      31 Jan  1 00:00:00 sdate.s
215 s-rwrw  1 root     231 Jan  1 00:00:00 sd.s
213 l-rwrw  2 root   16448 Jan  1 00:00:00 .core
```

Cf. `chmod(1)`, **HISTORY** for the access modes available, but the mode format consists of

l/s **l** if "large file" (file at least 4kB in size, there's another i-node with extents), else **s** (small)
d/u/x/- **d** if directory, else **u** if set-UID, else **x** if executable (there's one bit for this), else **-** (note that there are no device nodes yet, and devices are simply regular files with i-node ≤ 40)
{r|-}{w|-} access mode for owner
{r|-}{w|-} access mode for others

Version 4 AT&T UNIX sees a **SYNOPSIS** of

ls [-ltasdru] name ...

with **-ru** as present-day (access times are new in Version 4 AT&T UNIX). The status in **-l** is now just about the same as present-day: **d/-c/b** depending on file type, and **rwX** for each of owner, group, other (owner and group **x** become **s** if the appropriate set-ID bit is set; there is no set-other-ID/sticky bit). While the system freshly *has* groups, they are not listed with **-l**, or generally used at all.

Version 5 AT&T UNIX sees a **SYNOPSIS** of

ls [-ltasdruiF] name ...

-l now stuffs the device number as *major, minor* into the size column for devices, as present-day, (it's unclear if this happened in Version 4 AT&T UNIX; this also means that for directory blocks-used totals (and **-s**) the device numbers are counted as sizes as well). With the advent of **-i**, **-l** no longer lists it by default. Groups continue to be unused, but now have a field in `passwd(V)` (always set to **1**). The time format with **-l** is "Jul 4 16:08", as present-day for recent files (it's possible it was shortened earlier). **-i** may be combined with **-l**, but **-s** still can't.

-f is described as turning off **-ltsr** (effectively allowing only **-i**), enabling **-a**, and *forcibly* reading each *file* as a directory. This is apt, because directories contain only the i-node numbers and names of the files contained within. Thus, **ls -fi bkp** would let you read an old directory listing created with **cp . bkp**.

Version 6 AT&T UNIX sees a **SYNOPSIS** of

ls [-ltasdruiFg] name ...

and the addition of **-g**, replacing the owner name with the group name in the **-long output format** (but *not* enabling it), corresponds to the overall implementation of groups (and `group(V)` now appears). Undocumented, this is the first system that differentiates between old and recent files for date formatting, as present-day: "Jul 4 16:08" for recent ones and "Jul 4 2024" for old ones; the comment says *"/* 6 months ago */* but it's $245 \cdot 2^{16}$ seconds which is around 185 days and 20 hours, which isn't 6 months. Admittedly, this implementation considers "6 months" to be 180 days, which is also not 6 months, so.

Also,

The last character of the mode is normally blank but is printed as ```t``` if the 1000 bit of the mode is on.

the last character of the mode is actually normally the execute for others bit, and indeed **t** instead replaces the space *after* the mode; "the 1000 bit" is sticky (but real. see `chmod(1)`, **HISTORY** for how the sticky bit works on this system).

Version 7 AT&T UNIX sees a **SYNOPSIS** of

ls [-ltasdrucifg] name ...

with **-c**, as present-day, and reconsiders "6 months" as 180 days (largely owing to the fact that it has 32-bit integers now). **-s** can be combined with **-l**, as present-day. The **t** status is moved into the other-execute character, as present-day.

The BSD

1BSD patches Version 6 AT&T UNIX **ls** into **ls** (VI) synthesised as

ls [-ltasdrufm1cq] name ...

Note the removal of **-g**: 1BSD groups, rather than being an additional credential like modern groups, hard-partition users into group *GID*/subuser *UID* (two users are the same if both the *UID* and *GID* are the same). **-n** formats the credentials as "*uidugidg*".

This system is the origin of most of the modern output behaviour: **-q** is as present-day (except tabs and newlines aren't escaped), and set by default if the standard output stream is a teletype. **-c** is like **-C**; **-xm1** are also as present-day. Additionally, if **ls** is invoked as a name that ends with an 'l' then this is overridden and **-m** is implied, and if it ends with neither 'l' nor 's' then the default turns from **-c** to **-x**. */usr/ucb/l* is linked to */usr/ucb/ls* to take advantage of this. The *output-width* is always **80** (this system predates environment variables). Passed *files* are always listed in **-x** direction, even if directory entries are listed in **-c** direction.

1BSD quota files (character devices with device number **-1**, canonically ".q") are typified as **q** and the size column is "*used/max*" (512-byte blocks?).

2BSD upgrade/src/ls.c, which is a "supplement" "for version 6 systems.", reverts to a modern-style group model (the group name is not written, **-n** just writes the UID). The file mode sees group **rwX/rws** as well. **-A** is new and as present-day (except it's the default if running as root).

2BSD's **ls(UCB)** reimplements 1BSD's **ls** output modes (**-cmq1**, incl. autodetection, but with **-x** equivalent to **-c**) into Version 7 AT&T UNIX **ls**, and also adds **-An**, as present-day.

3BSD carries unchanged Version 7 AT&T UNIX in */bin/ls* and 2BSD in */usr/ucb/ls* (and */usr/ucb/l*), with **-x** re-added (but not autodetection by 's')

4.0BSD */usr/ucb/ls*, synthesised as

ls [-abcdfgilmqrstux1CFR] name ...

-c is renamed to **-C**, with **-c** now being "use *st_ctim*" (both as present-day). **-R** is as present-day (likewise overridden by **-d**). **-A** is still there, just undocumented.

-b is overridden by **-q** and always escapes as *\ooo*. **-F** decorates directories as */* and executable files as ***. *Except* if the UCB macro is somehow unset, in which case directories are decorated as *[dir]* and executables as **exe**; the purpose, use, or origin of this is unclear.

The supported alias menagerie grows to

```
ll      l      lx      lf      lr
ls -l   ls -m   ls -x   ls -F   ls -R
```

(except for **ll**, matched by the last two bytes, these are all matched by the last byte); only **l** is installed.

Multiplexed block/character special files (*S_IFMPB*, *S_IFMPC*), added in 3BSD, are typified as **M** and **m**, respectively, and the device numbers are formatted as for normal devices. They were treated like regular files in 3BSD.

4.2BSD sees *one* **ls** implementation, and a completely new one at that.

ls [-acdfgilqrstu1ACLFR] name ...

this amounts to losing **-bmX** and gaining **-L** (this is the first system with symbolic links). **-s** is reimagined as "give size in kilobytes of each file" (this is also the first system with *st_blocks*, so this now uses that); the same applies to the directory total. **-L** is as present-day, and similarly **-l** stops symbolic links to directories from being followed, but without it they are *always* followed? **-q** now hides *all* nonprintables. **-F1** flags symbolic links as "*filename@ -> target*". **-F** now produces */=@**; sockets and symbolic links are typified as present-day as well. *S_IFMPB* and *S_IFMPC* disappear from the system altogether. **-g** now *adds* the group name/ID instead of replacing the owner.

4.3BSD uses the current width of the teletype attached to the standard output stream (`tcgetwinsize(3)`), if any, instead of hard-coding **80**.

4.3BSD-Tahoe formats set-user- and -group-ID bits and the sticky bit in the file mode, as present-day.

4.3BSD-Reno sees a **SYNOPSIS** of

ls [**-1AaCcdFfgikLlqRrstu**] [file ...]

so only **-k** is new. **-s** and directory totals are reported in **512**-byte blocks, unless **-k**, which re-scales them back to kilobytes. Six months, for the purposes of **-l** time, is now *182* days. Symbolic link behaviour is now to follow for top-level *files* if **-L** or neither of **-ld**. And to not follow for **-R** ever? The *output-width* is the width of the teletype attached to the standard output stream, or `$COLUMNS`, or **80**, in that order.

4.4BSD Lite1 sees a **SYNOPSIS** of

ls [**-ACFLRTacdfiloqrstul**] [file ...]

meaning **-gk** removed and **-To** added. **-g** is actually accepted, but ignored (for `/* Compatibility with 4.3BSD. */`). **-T** changes the **-l** time format to the full "Jul 4 16:08:03 2024". The manual claims compatibility with IEEE Std 1003.2 ("POSIX.2") despite checking `$COLUMNS` *after* the output teletype width.

-o lists file flags (or "-") before the size in the **-long output format**. File flags (`chflags(2)`, `chflags(1)`) are new in this system, and rendered as a comma-separated list of

Macro	String	Settable by	Behaviour
UF_APPEND	uappnd	owner	O_APPEND only
UF_IMMUTABLE	uchg	owner	O_RDONLY only
UF_NODUMP	nodump	owner	dump(8) will ignore this by default
SF_APPEND	sappnd	root	O_APPEND only
SF_ARCHIVED	arch	root	no effect; unused; undocumented
SF_IMMUTABLE	schg	root	O_RDONLY only

Symbolic link behaviour is sane and almost as present-day, except if none of **-Filst** *all* top-level symbolic links are followed. **-F** typifies FIFOs as **|**, even though they were fresh in 4.3BSD-Reno. The block size for **-s** and directory totals is `$BLOCKSIZE` via `getbsize(3)` — case-insensitive

[*base*][**KMG**]

with an integer *base*, units in powers of **1024**, clamped to [**512**, **1G**] with a warning, and a default of **512**.

4.4BSD-Lite2 adds white-outs (as part of `mount_union(8)` support), typified as **w**, decorated as **%**, hidden by default unless **-W**. The new owner-settable `UF_OPAQUE/opaque` flag makes a directory "opaque when viewed through a union stack".

System V

AT&T System III UNIX sees a **SYNOPSIS** of

ls [**-logtasdrucif**] names

where **-l** writes both the owner and group names, and **-og** become more like present-day, inasmuch as they enable **-l**, but they are treated as separate modes (rather than flags for **-l**). Device numbers are no longer counted as sizes for totals and devices' **-s** is *0*. The file mode is exactly as present-day now (**sst** become **SST** if the corresponding *x* bits are clear; cf. `chmod(1)`). `fifo(7)s`, new in this system, are typified as **p**, as present-day. There's an undocumented **-b** flag, identical to **-l**.

AT&T System V Release 1 UNIX sees a **SYNOPSIS** of

ls [**-logtasdrucifp**] names

and **-p** is as present-day. Block sizes for directory totals and **-s** adapt to the system's supported filesystem (either *512* bytes, as previous, and *1024* bytes for the "new" filesystem, or if both are supported). The phantom **-b** flag is gone.

AT&T System V Release 2 UNIX sees a **SYNOPSIS** of

ls [**-RadCxmlnogrtnucpFbqisf**] [names]

-go are now as present-day. **-nCxmR** are as present-day. The *output-width* is `$COLUMNS`, else the user's teletype width (according to `$TERM` and the `terminfo(5)` database), but if it's <20 or >160 then it's reset to the default of 80. The width of each filename is counted as its byte length, and indeed

BUGS says

Unprintable characters in file names may confuse the columnar output options. so **-q** (which is as present-day) works well with them. **-b**, which *always* produces octal escapes, doesn't. **-F** produces / and * only.

AT&T System V Release 3 UNIX is the first system with a maximum passed-to-**open()** path size, and thus extends this buffer to MAXNAMLEN (**512**, à la PATH_MAX). There's still not a bounds check, but it's a much-more-sensible size. The set-group-ID *but* group-not-executable status is changed from **S** to **I**, since this now indicates mandatory locking (cf. **chmod(1)**, **HISTORY**).

AT&T System V Release 4 UNIX sees a **SYNOPSIS** of

```
ls [-RadLCxmlnogrTucpFbqisf1] [names]
```

This is the first system with symbolic links; **-L** and "filename -> target". in **-l** are as present-day. Argument *files* which are symbolic links to directories are followed, as present-day, but this isn't disabled by **-lF**. **-F** now produces @ for symbolic links and produces decorations like this implementation.

-l is as present-day, required since **-C** is now the default if the standard output stream is a teletype (or if **ls** was executed as **lc**). Rather than the terminfo(5) database, the current-teletype-at-standard-output-stream's current width (tcgetwinsize(3)) is consulted, as present-day. This system also grows *st_blocks*, so these are used directly instead of deriving from the size. This **stat(2)** field is in units of **512** bytes (as present-day), so the scaling is fixed at **512**.

XENIX

XENIX has a S_IFNAM file type, with S_INSEM (semaphore) and S_INSHD (shared memory (sometimes "shared data")) subtypes (also sometimes IFREM which is.. not? exposed to userspace?). **ls** typifies these as **s** and **m**, respectively, and all others default to -. XENIX is variably derived from AT&T System III UNIX and later systems (but this interface was largely supplanted by **sysvipc(7)**).

SunOS

SunOS 1 is derived from 4.2BSD and doesn't indicate any differences, with a **SYNOPSIS** of

```
ls [-acdfgilqrstu1ACLFR] name ...
```

SunOS 2 describes **-A** as "same as **-a**", with a **SYNOPSIS** of

```
ls [-acdfgilqrstu1ACLFR] name ...
```

SunOS 3 is the first version with a source dump available, so: the "file is from the future" test for using the old date format is relaxed to be for an *hour*, rather than a minute, from now. **fifo(7)**s are typified **p** but *not* **-F**-decorated. Symbolic links in **-lF** mode are interrogated to get the destination decoration, and only the target is decorated. Symbolic links to directories given directly as *files* are no longer followed.

This **ls** lives in **/bin/ls**, but a AT&T System V Release 2 UNIX-derived **ls** (with symbolic link support (copied from the normal **ls**), **-L**, sockets typified as **s** and **=**) is also shipped, in **/usr/5bin/ls**, now in **ls(1V)**, synopsised as

SYNOPSIS

```
ls [-acdfgilqrstu1ACLFR] filename ...
```

SYSTEM V SYNOPSIS

```
/usr/5bin/ls [-abdcfgilmnopqrstuxCLFR] filename ...
```

SunOS 4 (Solaris) synopsises **lses** as

SYNOPSIS

```
ls [-aAcCdFfGgILlQrRstu1] filename ...
```

SYSTEM V SYNOPSIS

```
/usr/5bin/ls [-abcCdFfGgILlmnopqrRstux] filename ...
```

the astute reader will note that this simply re-sorts the flags, and doesn't change the list. This is the first system that documents the six-month date format threshold and the formats themselves (as "month date year" and "month date time", admittedly). The manual freshly mentions using the LC_CTYPE locale category, but it doesn't say what for.

`/usr/ucb/ls` is loosely rebased on 4.3BSD. `/usr/5bin/ls` is unchanged.

AT&T System V Release 4 UNIX includes this implementation in `/usr/ucb/ls` (but with the ACL mode column removed).

SunOS 5 (Solaris 2) adds **+** to the mode if non-default ACLs are present (much like present-day) and removes the **T** sticky-bit special case.

SunOS 5.7 (Solaris 2.7) adds doors (thus, **D**, **>**).

The maximum total path length (for example, `dire/ctory/file` if `ls dire/ctory` and it contained `file`) is fixed in all implementations, and longer names are truncated and end up being unstable. Version 1 AT&T UNIX has a 34-byte buffer (the maximum filename size is 8 bytes); Version 4 AT&T UNIX has a 14-byte maximum filename, and Version 5 AT&T UNIX `ls` has a 100-byte static buffer (this includes the NUL terminator, but doesn't include a bounds check, so longer ones probably work too, until they trample something else) (it's unclear if this was changed by Version 4 AT&T UNIX as well); AT&T System III UNIX extends this by `DIRSIZ` (14). This isn't really an issue until **-R**, which will, depending on how the statics were laid out by the linker, produce more and more garbage until it segfaults when it encounters a deep directory tree. 4.2BSD checks the bounds for its `BUFSIZ` (1024)-sized buffer, which matches so it only suffers its `MAXPATHLEN` (1024). 4.4BSD uses `fts(3)`, which spills its implementation quirks into **-R** limits: the default symlinks-unfollowed (`FTS_PHYSICAL`) mode can walk arbitrarily deep by `chdir(2)`ing, but the symlinks-followed (`FTS_LOGICAL`) mode doesn't do this and grows a single buffer ad infinitum.

Standards

System V Interface Definition Issue 2 ("SVID2"), includes AT&T System V Release 3 UNIX `ls(1)` almost-verbatim as `ls(BU_CMD)` ("`ls` - list contents of directory") with a **SYNOPSIS** of

```
ls [ options ] [ file ... ]
```

defining block sizes as **512** and removing **-b** (most likely the System V Interface Definition ("SVID"), includes AT&T System V Release 2 UNIX `ls(1)` as well, but it doesn't appear to be extant)).

X/Open Portability Guide Issue 2 ("XPG2") includes this `ls`, with **-CFRglnmosx** aptly shaded OF ("Output format incompletely specified") and **-nx** less-aptly shaded UN ("Possibly unsupportable feature."). **-f** is in full swing and given with a straight face. **-F** *requires* `/*`, but *allows* other decorators for other file types. Similarly, other file type characters are allowed in the mode, and the **I/T** split is resolved by allowing anything for implementation-specific permissions. The **APPLICATION USAGE** astutely notes that, in fact, not all implementations write **-s** and directory totals in 512-byte blocks (indeed, in the case of SunOS 2, a single implementation can provide `lses` with *two* block sizes).

X/Open Portability Guide Issue 3 ("XPG3") defines the filename sort order to the present-day "current locale's collation order".

IEEE Std 1003.2 ("POSIX.2") includes

```
ls [-CFRacdilqprtul] [file ...]
```

with most text clearly derived from the same source. Symbolic links are entirely unmentioned. This means the addition of **-l** (as present-day), and removal of **-fgmnopsx** (**-x** "provided only limited increased functionality over the **-C** option") (**-m** "was not provided because similar functionality (easier to parse for scripts) can be provided by the `echo` and `printf` utilities") (**-s** because "The number of disk blocks occupied by the file that it reports varies depending on underlying file system type, block size units reported, and the method of calculating the number of blocks.") but the directory blocks-used total is in 512-byte-blocks (even though they just said the value itself is not really useful or obtainable portably), because the BSD variably uses 512 and 1024 but AT&T System V UNIX uses 512. This discrepancy persists to this day.

The BSD `ls` provides a **-A** option (like **-a**, but dot and dot-dot are not written out). The small difference from **-a** did not seem important enough to require both.

-F mandates `/*`. The description for **-a** allows hidden filenames to be shown, even if **-a** wasn't given, if some "implementation-defined condition" is met (clearly to appease 2BSD **-A**). The entire format is fully specified; the **-l** mode allows an "<optional alternate access method flag>" to provide for the ACL **+**. **-t** is for the first time described as "first time, then filename", as opposed to "time instead

of filename" heretofore.

X/Open Portability Guide Issue 4 ("XPG4") (The Single UNIX Specification ("SUS")) merges IEEE Std 1003.2 ("POSIX.2"), for a **SYNOPSIS** of

```
ls [-CFRacdilqrtu1] [-fgmnoptsx] [file...]
```

with **-fgmnoptsx** shaded EX ("Extension.", equivalent to modern-day XSI shading). **-s** says that "The block size is implementation-dependent."; the **FUTURE DIRECTIONS** warn that because **-s** "cannot be used portably; it will be withdrawn from a future issue".

Version 2 of the Single UNIX Specification ("SUSv2") replaces the "will" with "may" and further threatens alignment with IEEE Std 1003.2b ("POSIX.2b": Shell and Utilities — Amendment). To this end,

IEEE Std 1003.2b ("POSIX.2b": Shell and Utilities — Amendment) mandates that the **Synopsis** be modified to

```
ls [-CFRacdilqrtu1] [-H | -L] [file...]
```

and defines symbolic link behaviour as present-day: top-level *files* are followed if **-H** or **-L** or (*file* is a symbolic link to a directory *and not* **-dF1**); directory entries and everything below is followed if **-L**. Loops must be detected (but both hard-exit and soft failure are allowed). **-F** decorates symbolic links as **l**, but doesn't specify where to decorate "*filename -> target*".

Version 3 of the Single UNIX Specification ("SUSv3") merges IEEE Std 1003.2b ("POSIX.2b": Shell and Utilities — Amendment) directly for a **SYNOPSIS** of

```
ls [-CFRacdilqrtu1] [-H | -L] [-fgmnoptsx] [file...]
```

and allows additional **-F** decorations.

IEEE Std 1003.1-2008 ("POSIX.1") sees a **SYNOPSIS** of

```
ls [-ikqrs] [-glnO] [-A|-a] [-C|-m|-x|-1] \
    [-F|-p] [-H|-L] [-R|-d] [-S|-f|-t] [-c|-u] [file...]
```

(with only **-go** shaded XSI, meaning **-fmnoptsx** are baseline; for sanity. **-f** becomes "don't sort; also sorry there's a bunch of flags this toggles (we may remove that latter bit later)"), which amounts to an addition of **-A** —

Earlier versions of this standard did not describe the BSD **-A** option (like **-a**, but dot and dot-dot are not written out). It has been added due to widespread implementation.

— and **-k** ("to provide a way for the **-s** option to be used portably"; the asymmetry of 512-byte blocks for directory totals vs implementation-defined for **-s** seems to be by design here) and **-S** indicates that it standardises common practice (and discusses at length why, in fact, the documentation for what **-S** said it did on those systems was wrong for no reason).

IEEE Std 1003.1-2024 ("POSIX.1") clarifies that **-R** only visits subdirectories that weren't hidden by lack of **-Aa** (AIX, supposedly, visits every subdirectory) and that file type decorations for symbolic links in the **-long output format** — "*filename -> target*". — may go after just the *filename*, just the *target*, or both. The *s* mode *type* and **=** decoration is enshrined for sockets.