

inodes

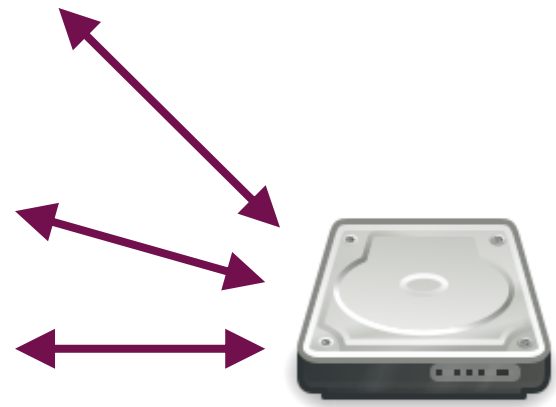
COMS10012 Software Tools

file system



disks: block devices

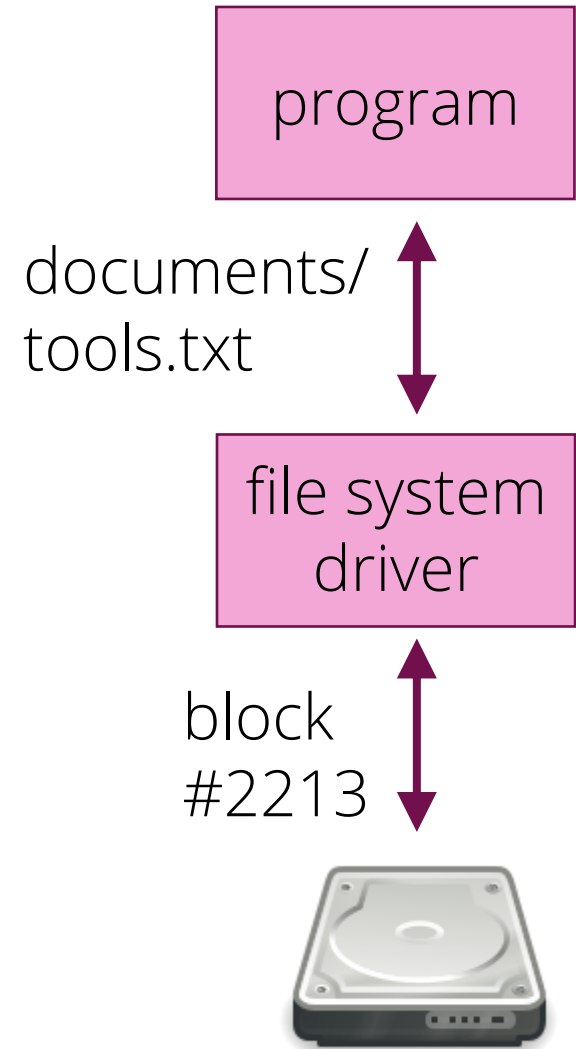
- What is the number and size of your blocks?
- Read block N into buffer.
- Write block N from buffer.



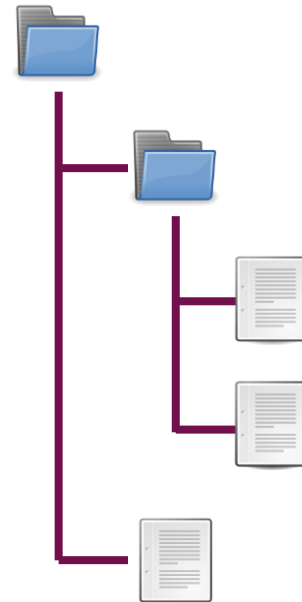
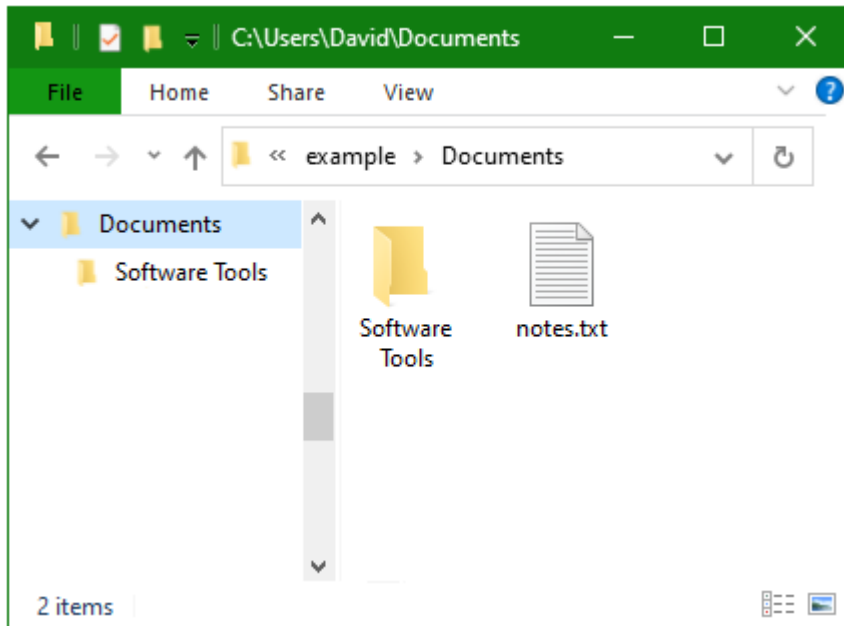
e.g.: `int write_blk(int blk_id, char* buffer);`

file systems

- file names
- different file sizes
- folders
- permissions
- journaling etc.



file tree



Documents
Software Tools
lectures.txt
planning.txt
notes.txt

Documents/software tools/planning.txt

paths

POSIX: path separator is forward slash /

A path starting with / is called absolute and starts at the filesystem root.

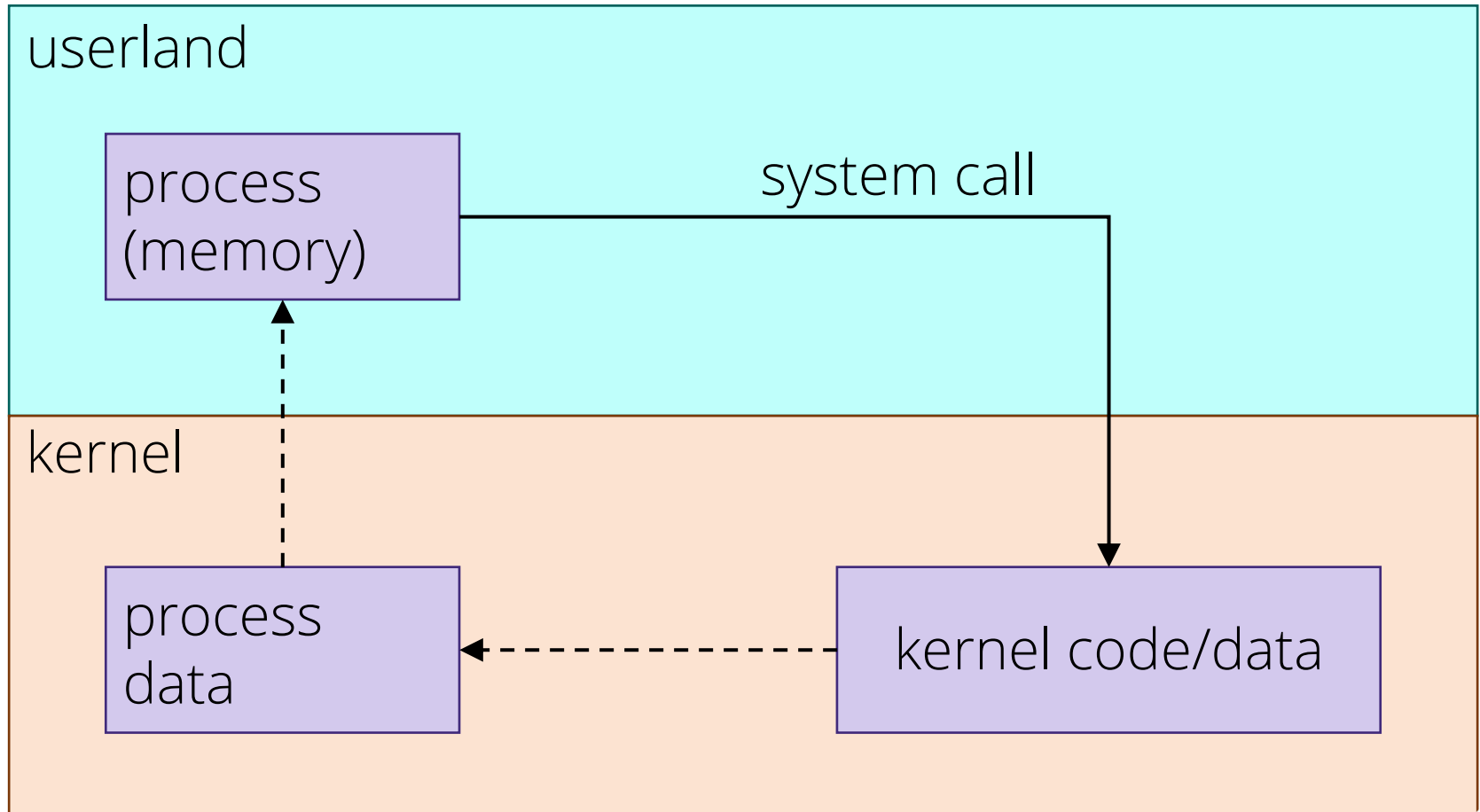
A path not starting with / is called relative and generally means from the working directory.

shortcuts: . (dot) = working dir, .. = parent dir

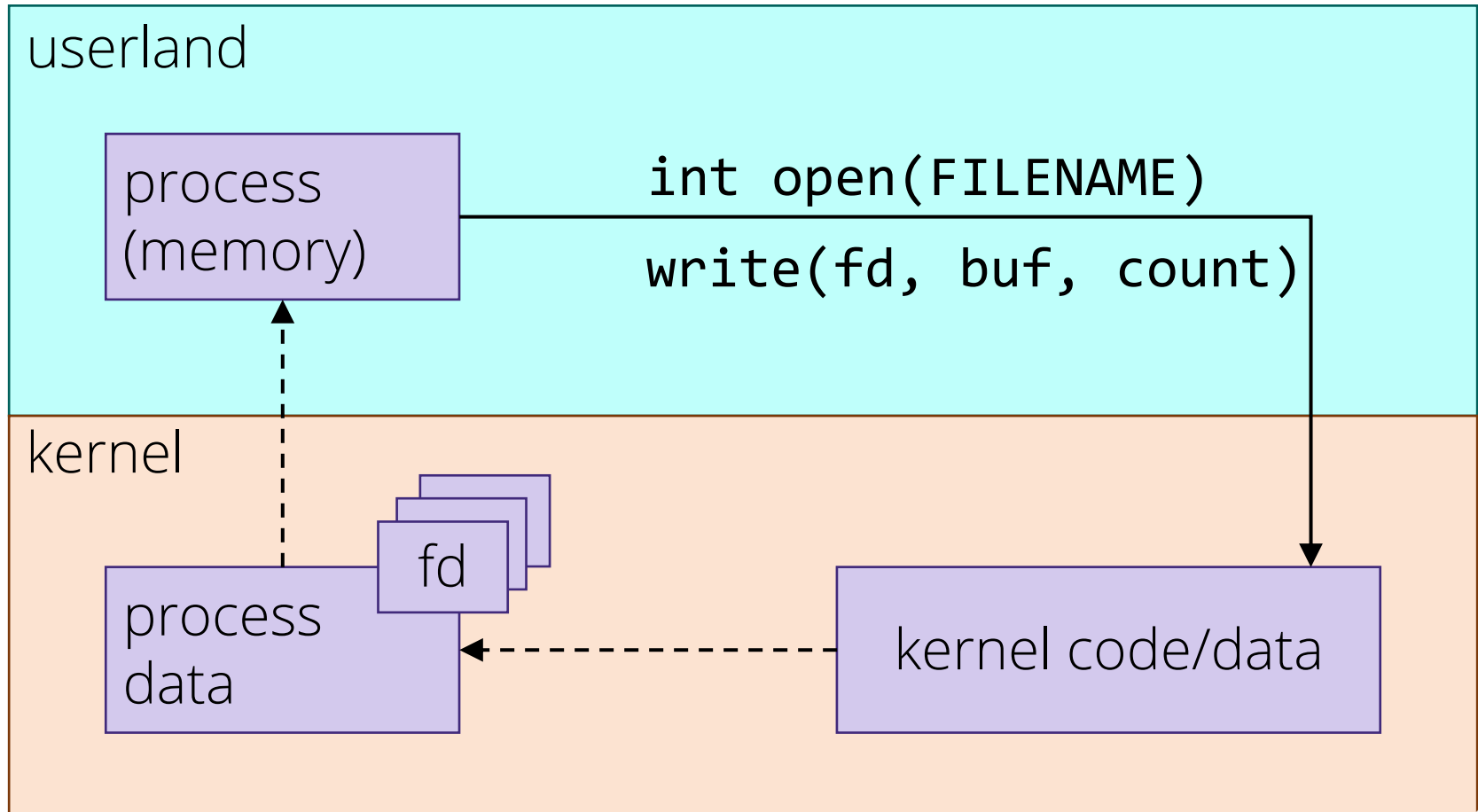
file descriptors



memory



file I/O



C and POSIX

C - `stdio.h`

```
FILE* fopen(const char *pathname, const char *mode)
size_t fread(void* buf, size_t size,
              size_t nmemb, FILE* stream)
int fclose(FILE* stream)
```

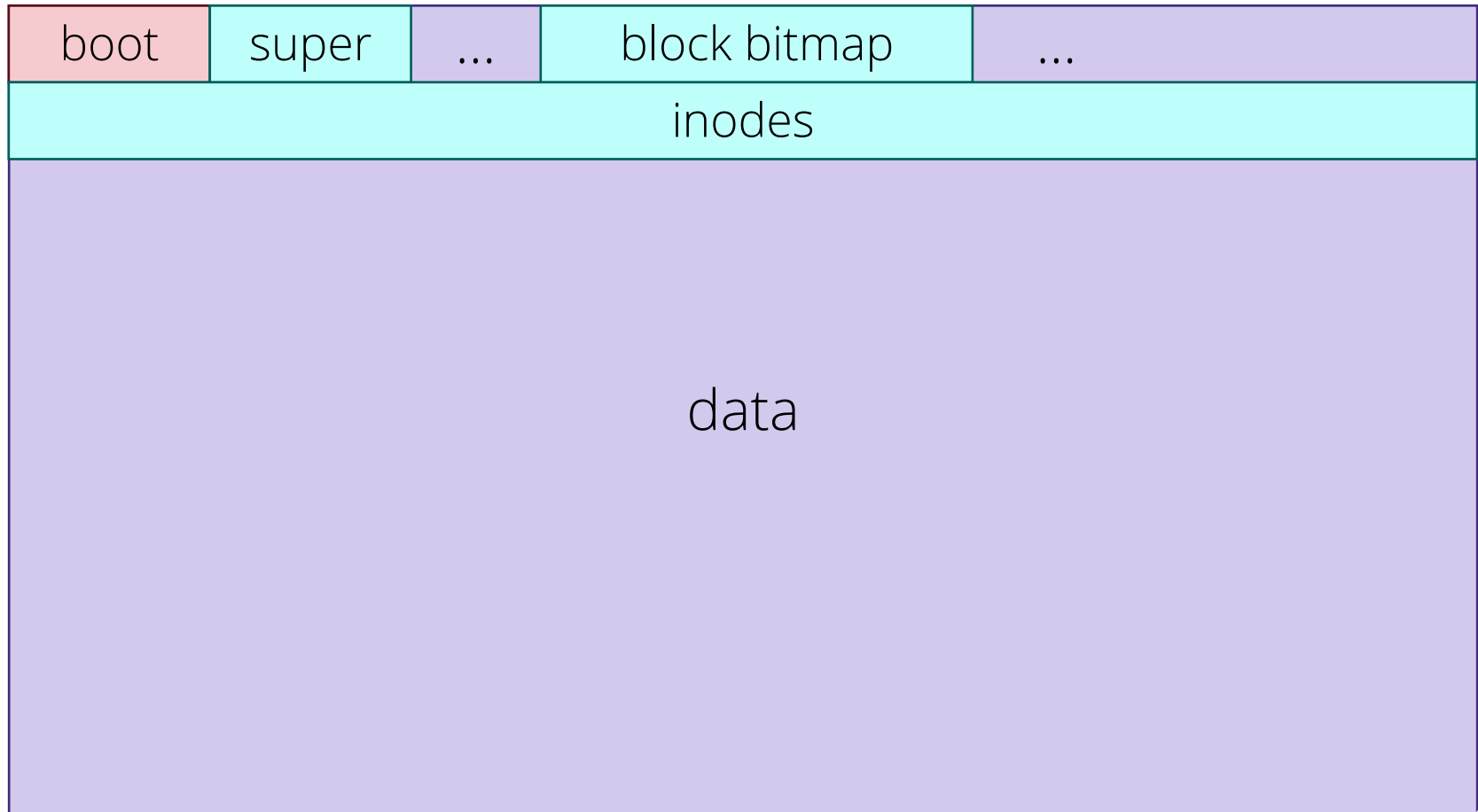
POSIX - `sys/fcntl.h`

```
int open(const char *pathname, int flags)
ssize_t read(int fd, void *buf, int count)
int close(int fd)
```

inodes



disk layout (extN filesystem)



inodes

inode: C struct for each file with

- unique id
- owner, group, permissions
- metadata (time of last access etc.)
- link count
- pointers to data blocks



data blocks

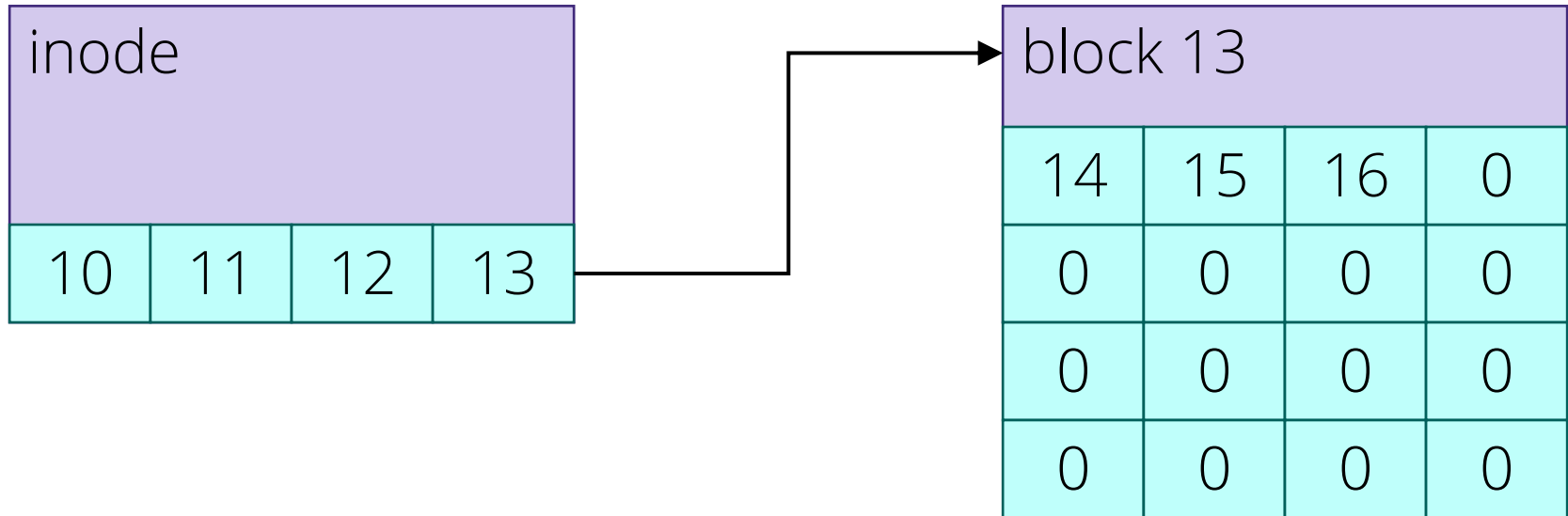
inode			
10	11	12	0

data block pointers

disk

0	1	2	3	4	5	6	7	8	9	10
10	11	12	13	14	15	16	17	18	19	20

data blocks



disk

0	1	2	3	4	5	6	7	8	9	10
10	11	12	13	14	15	16	17	18	19	20

inodes

```
alpine310:~$ ls -lai /bin
total 2396
786433 drwxr-xr-x    2 root    root          4096 Nov 13 09:15 .
      2 drwxr-xr-x   23 root    root          4096 Nov 13 09:10 ..
786436 lrwxrwxrwx    1 root    root           12 Nov  5 04:24 arch -> /bin/busybox
786437 lrwxrwxrwx    1 root    root           12 Nov  5 04:24 ash -> /bin/busybox
786438 lrwxrwxrwx    1 root    root           12 Nov  5 04:24 base64 -> /bin/busybox
786483 -rwxr-xr-x    1 root    root       735488 Jul 11 10:23 bash
786439 lrwxrwxrwx    1 root    root           12 Nov  5 04:24 bbconfig -> /bin/busybox
786481 ---s---x--x    1 root    root       14072 Aug 24 10:22 bbsuid
786454 -rwxr-xr-x    1 root    root      845384 Aug 24 10:22 busybox
786440 lrwxrwxrwx    1 root    root           12 Nov  5 04:24 cat -> /bin/busybox
```


links

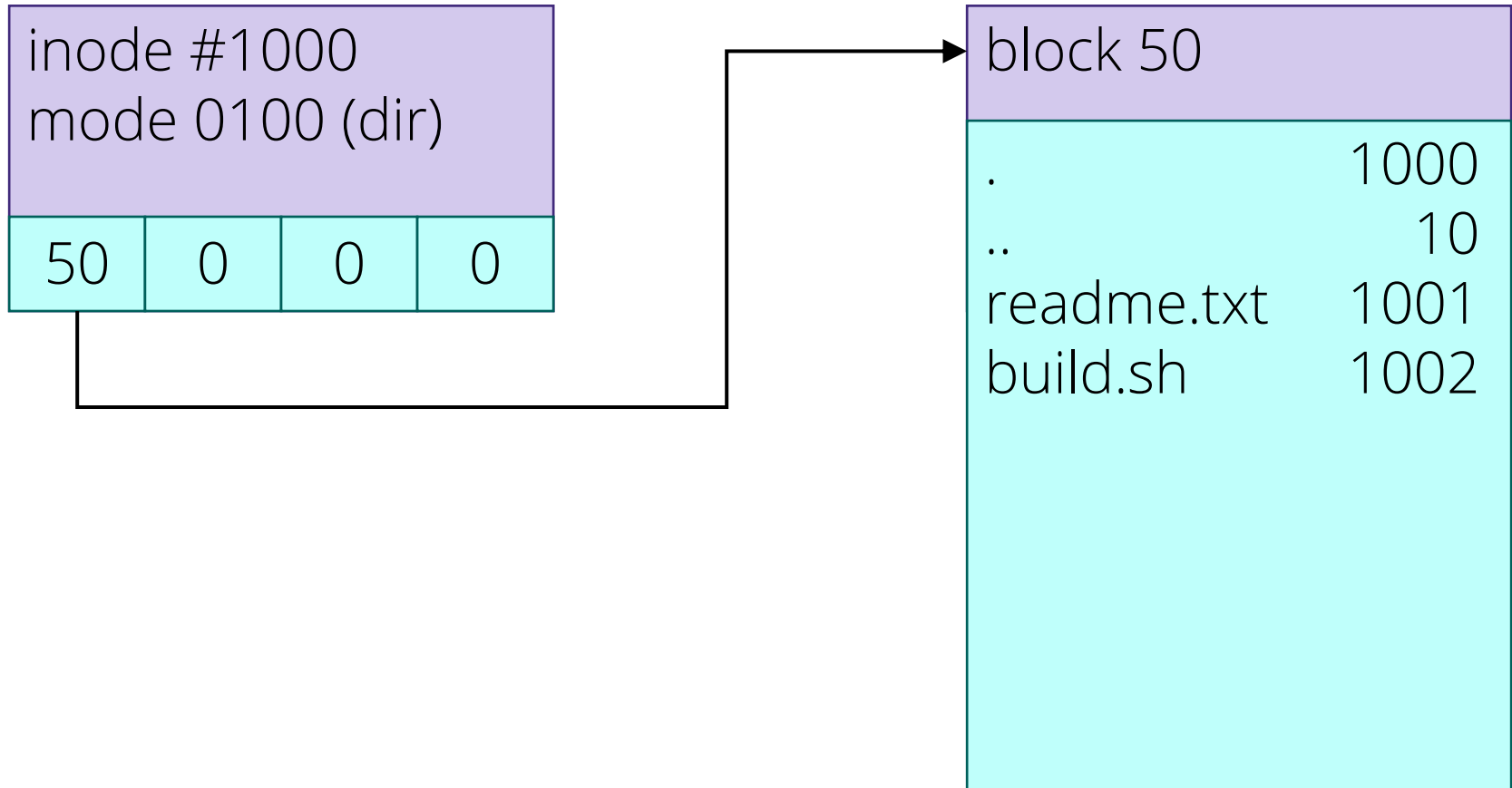


names

There is no filename in an inode.

"Files do not have names. Names have files."

directories



links

```
$ ls -li
```

```
100 -rw-r--r-- 2 user group 1024 ... file1
100 -rw-r--r-- 2 user group 1024 ... file2
101 -rw-r--r-- 1 user group 1024 ... file3
```

inode #100

mode: 1000 (file) / 0644
owner: user / group
links: 2

.	200
..	10
file1	100
file2	100
file3	101

file commands

touch FILE

creates an inode with one link
(assuming file does not exist)

rm FILE

decreases link count by one
if it reaches 0, data is freed

ln FILE NEW

creates a second link (NEW)
to an existing file

soft links

```
$ ln -s FILE NEW
```

Creates a soft link (mode 1010).

The data block for this link contains a filename.

Most operations on the link are redirected to the target filename.

This is how busybox binaries work.

