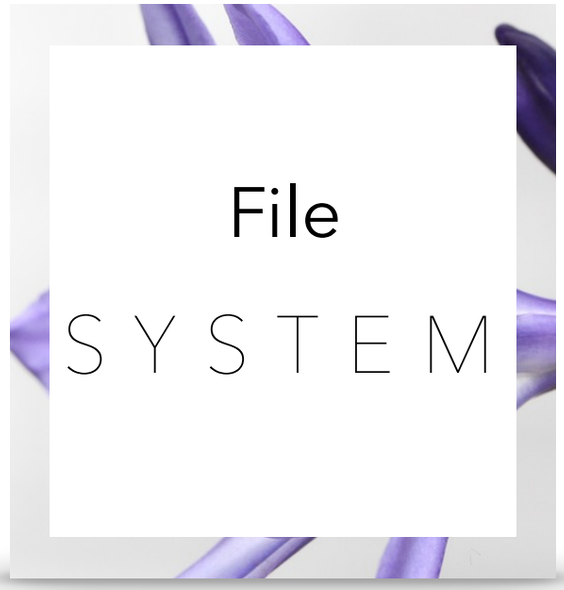


50.005 CSE

Natalie Agus
Information Systems Technology and Design
SUTD

WHAT IS A FILE?

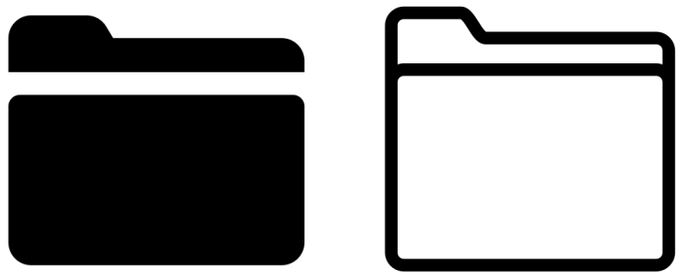
Abstract data type with two key attributes:

(1) **States / attributes** : file data and meta data, usage

(2) **Interface** : ‘methods’ to interact with the file

Basically, it is a **named** collection of related information that is recorded on secondary storage. **Not the content itself.**

Regular files



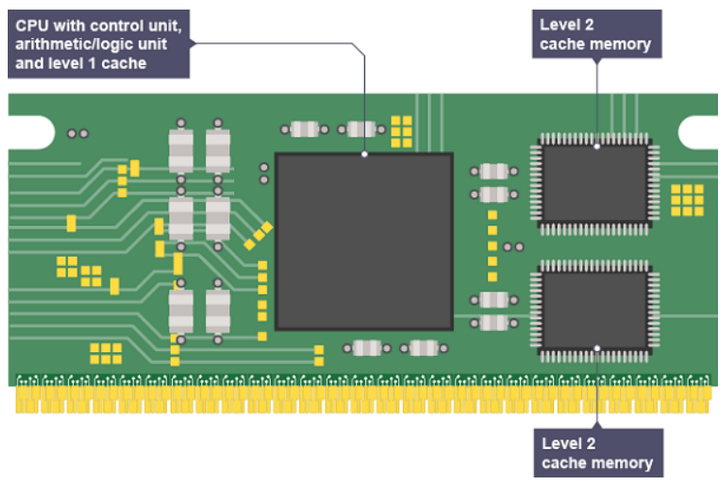
Directories or folder

WHERE ARE THESE FILES?

Depends

On Disk

On RAM



On Regs / Cache

FORMATS

NONE	SIMPLE	COMPLEX
Uninterpreted bytes	Lines (.txt), FLE, VLE	ELF format (binary executables)

WHO READS THEM?

USER APPS

SYSTEM PROGRAMS

KERNEL

file type	usual extension	function
executable	exe, com, bin or none	ready-to-run machine-language program
object	obj, o	compiled, machine language, not linked
source code	c, cc, java, pas, asm, a	source code in various languages
batch	bat, sh	commands to the command interpreter
text	txt, doc	textual data, documents
word processor	wp, tex, rtf, doc	various word-processor formats
library	lib, a, so, dll	libraries of routines for programmers
print or view	ps, pdf, jpg	ASCII or binary file in a format for printing or viewing
archive	arc, zip, tar	related files grouped into one file, sometimes compressed, for archiving or storage
multimedia	mpeg, mov, rm, mp3, avi	binary file containing audio or A/V information

FILE ATTRIBUTES

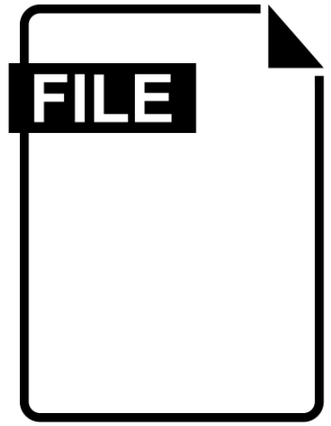
a.k.a metadata
a set of data that gives information about other data

Type

Protection

Size

id



Name

Location

User id, date, time

FILE INTERFACE

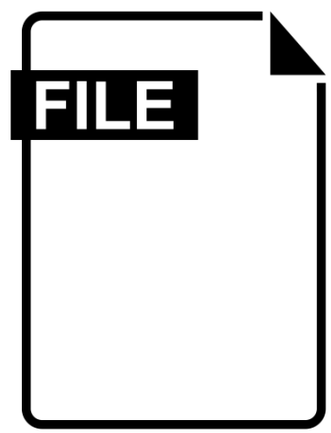
what can we do with a file?

Create

Open

Read

Delete



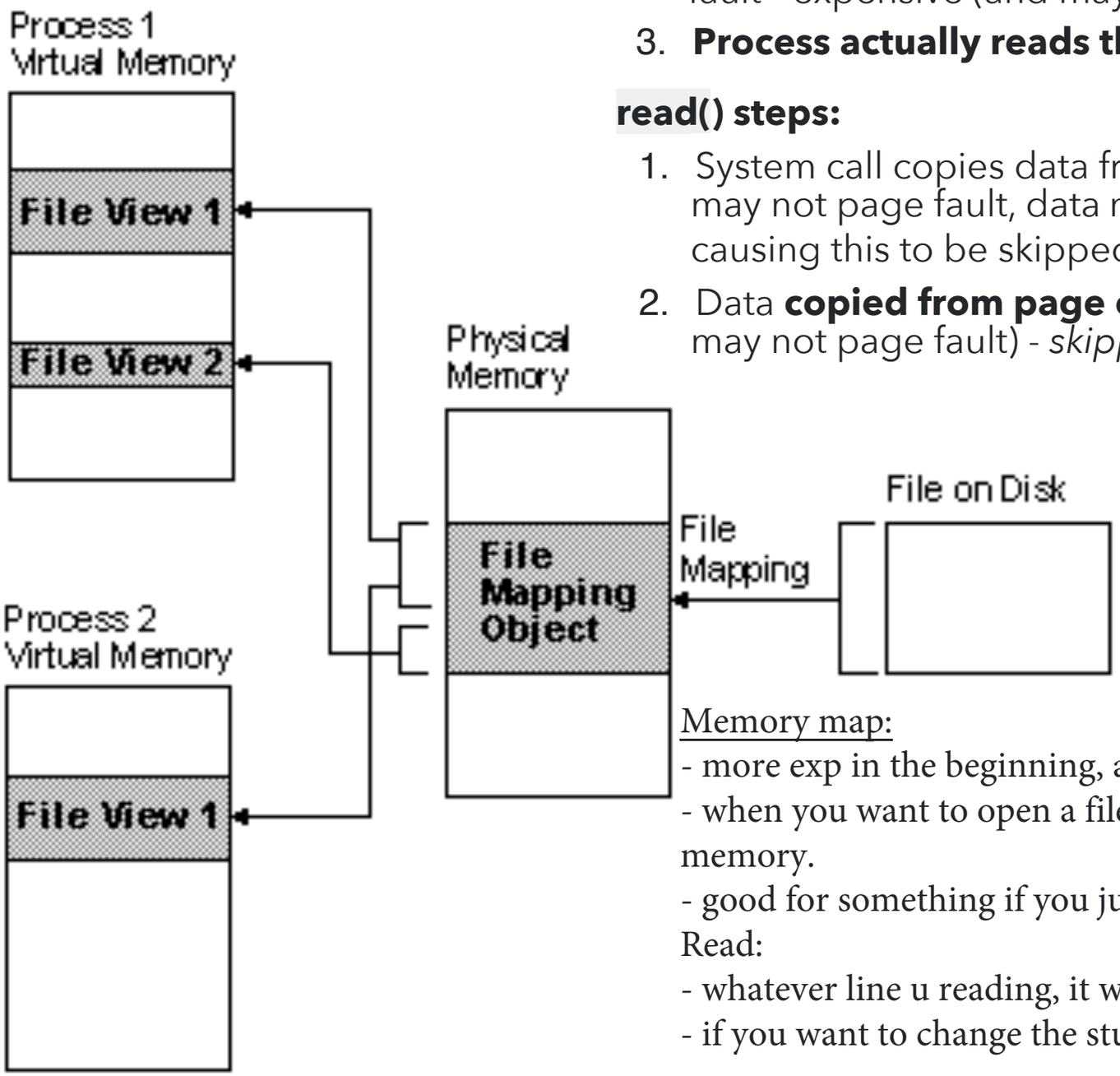
Truncate

Memory map

Write

lseek()

View: a portion of the file
Mapping: move from disk to RAM



Memory map steps:

1. System call to create virtual mappings - very expensive
2. Process accesses memory for the first time, causing a page fault - expensive (and may need to be repeated if paged out)
3. **Process actually reads the memory**

read() steps:

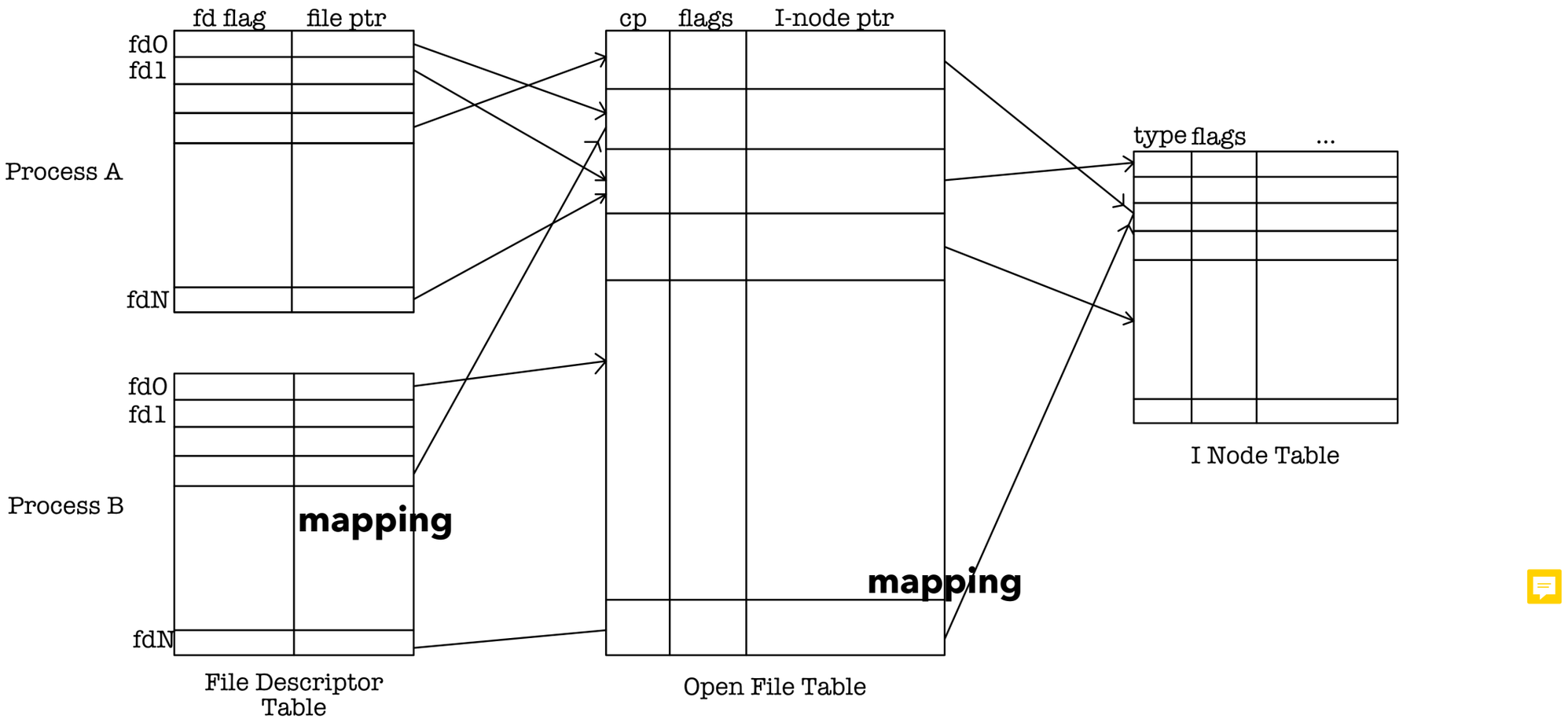
1. System call copies data from disk to page cache (may or may not page fault, data may already be in page cache causing this to be skipped)
2. Data **copied from page cache to process memory** (may or may not page fault) - *skipped in memory mapping*

Memory map:
- more exp in the beginning, after that can refer easily
- when you want to open a file, it moves the entire file over to the memory.
- good for something if you just want to read-only

Read:
- whatever line u reading, it will copy from disk to its space.
- if you want to change the stuffs

OPENING REGULAR FILES (UNIX)

```
90      int fd = open("/Users/natalieagus/Dropbox/50.005 Computer System Engineering/C_Codes/foo.txt", O_WRONLY | O_CREAT | O_TRUNC, 0644);
91
```



Each table **keeps track of reference count**. Entry is deleted when reference count is zero

Note: Open File Table isn't the only table pointing to I-node table. Files also point to I-node table.

CHILD'S FD TABLE

Child inherits entire fd table upon fork(), with the same state when fork() is called

After that child and parent process progress independently, with independent fd tables

STORING A FILE: DIRECTORIES

A directory is also a file that has the names of other files as its content.
In other words, a meta-data that organizes files in a structured name space

Traverse

Create

Rename

Search a file

List

Delete

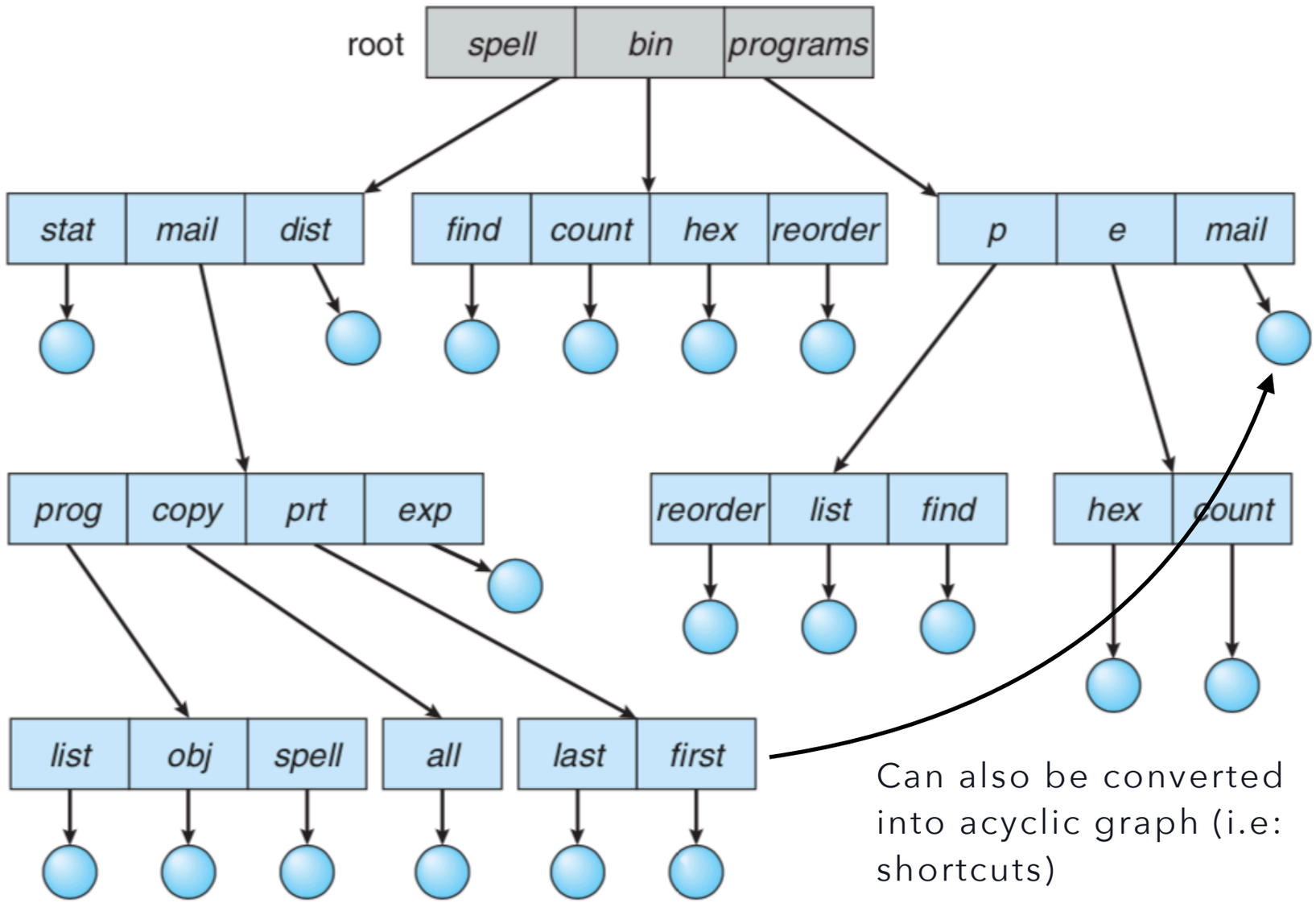
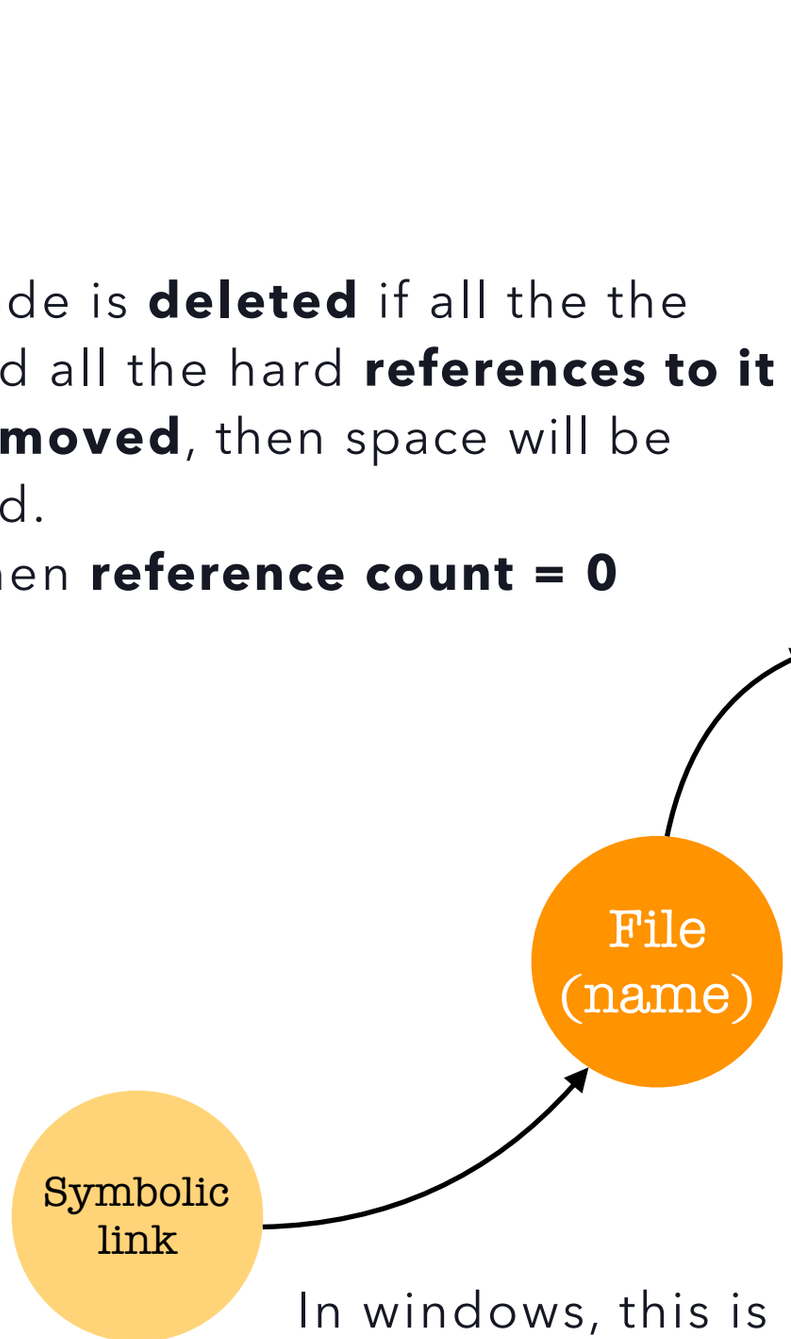


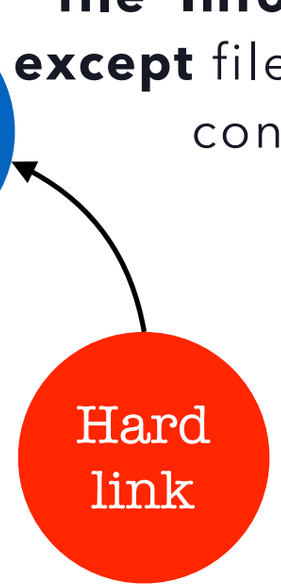
Figure 10.10 Tree-structured directory structure.

An inode is **deleted** if all the the file and all the hard **references to it are removed**, then space will be cleared.
i.e: when **reference count = 0**

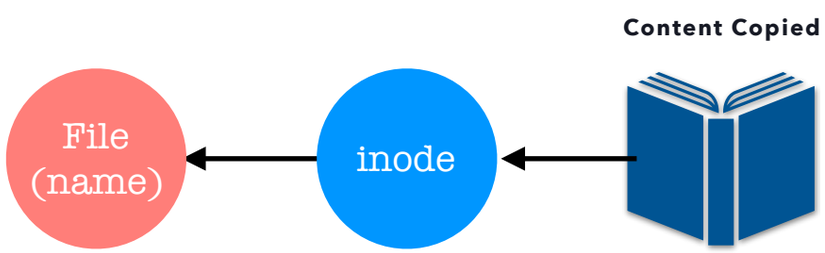


In windows, this is known as “shortcut”

UNIX “database” of file Information, except file name and content



Another name for the same content



Hard link != copy, a **copy** operation copies the entire content on secondary storage

CYCLIC DIRECTORIES

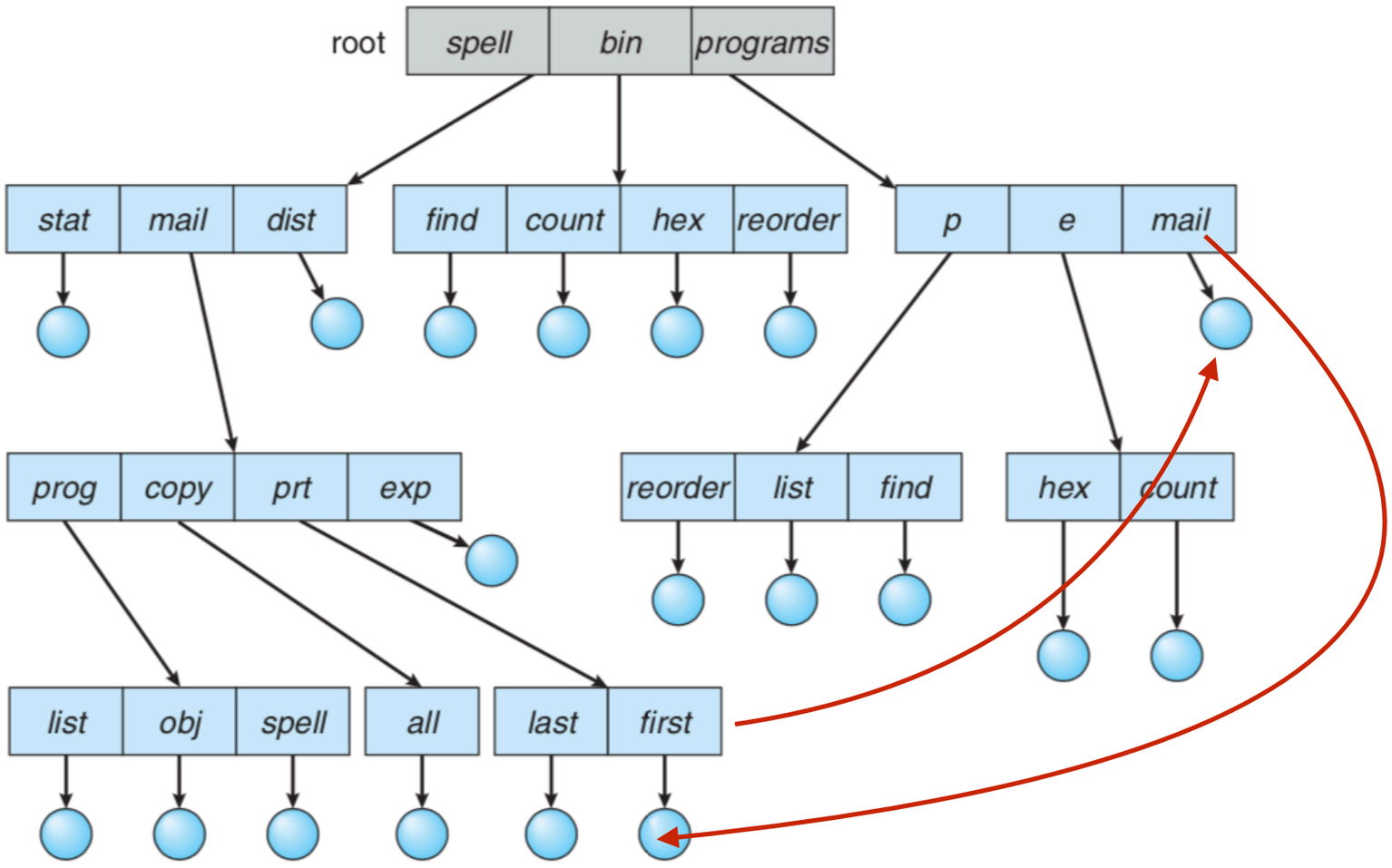


Figure 10.10 Tree-structured directory structure.

Possible, but BFS / DFS directory traversal may not terminate. Also, it will **self reference,** so reference count will never reach zero even if some root