

## Ken Thompson



Thompson (left) with Dennis Ritchie

# UNIX

## Dennis Ritchie & Ken Thompson

**Known for** Unix  
B (programming language)  
Belle (chess machine)  
UTF-8  
Plan 9 from Bell Labs  
Inferno (operating system)  
grep  
Endgame tablebase  
Go

**Known for** ALTRAN  
B  
BCPL  
C  
Multics  
Unix

In 1983, Ritchie and Thompson received the [Turing Award](#) "for their development of generic operating systems theory and specifically for the implementation of the UNIX operating system"

# Background

- ...
- Developed in C!

The greater part of UNIX software is written in the above-mentioned C language [6]. Early versions of the operating system were written in assembly language, but during the summer of 1973, it was rewritten in C.

# Goal

- **Simplicity!**



Perhaps the most important achievement of UNIX is to demonstrate that a powerful operating system for interactive use need not be expensive either in equipment or in human effort: UNIX can run on hardware costing as little as \$40,000, and less than two man-years were spent on the main system software. Yet UNIX contains a number of features seldom offered even in much larger systems. It is hoped, however, the users of UNIX will find that the most important characteristics of the system are its simplicity, elegance, and ease of use.

here. Most of them are engaged in applications such as the preparation and formatting of patent applications and other textual material, the collection and processing of trouble data from various switching machines within the Bell System, and recording and checking telephone service orders. Our own installation is used mainly for research in operating systems, languages, computer networks, and other topics in computer science, and also for document preparation.

# Outline

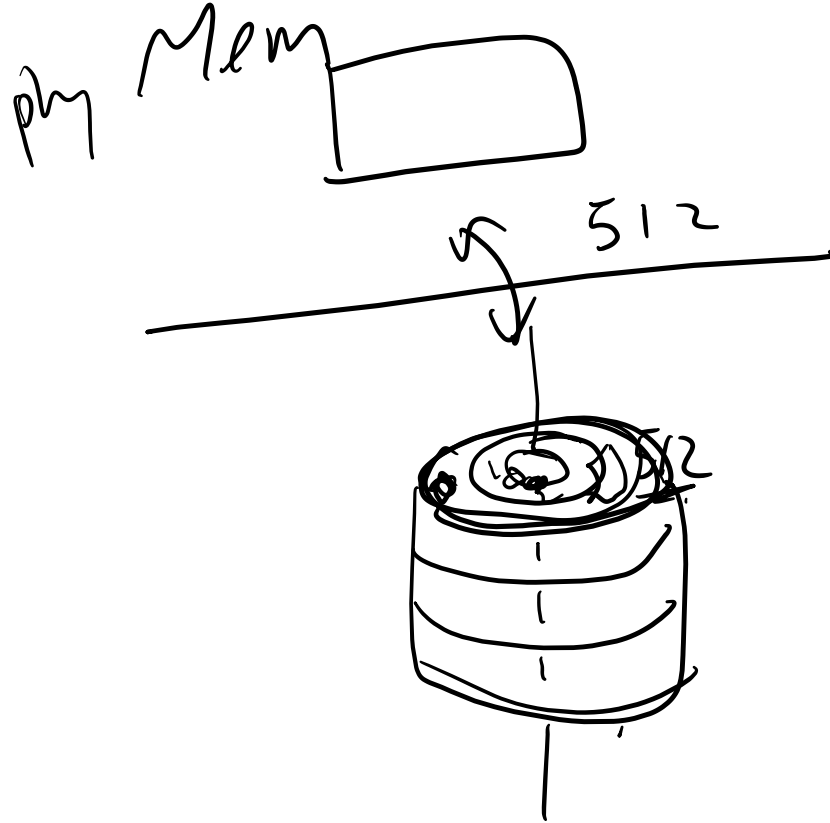
- Unix file systems
  - Files & directories
  - The implementation of files
    - How to locate a data block of a file?
    - Open, read, write, close, delete, link
  - Protection mechanisms
    - setuid
  - What helped its performance?
- Processes
- Shell

# Files `/xxx/yyy/zzz`

- Everything is a file
  - Regular files
  - Directories
  
- Devices

# Things related to a Unix file

- Path and name
- Owner, group, security setting
- Data
  - Data blocks (512B)
- Meta data



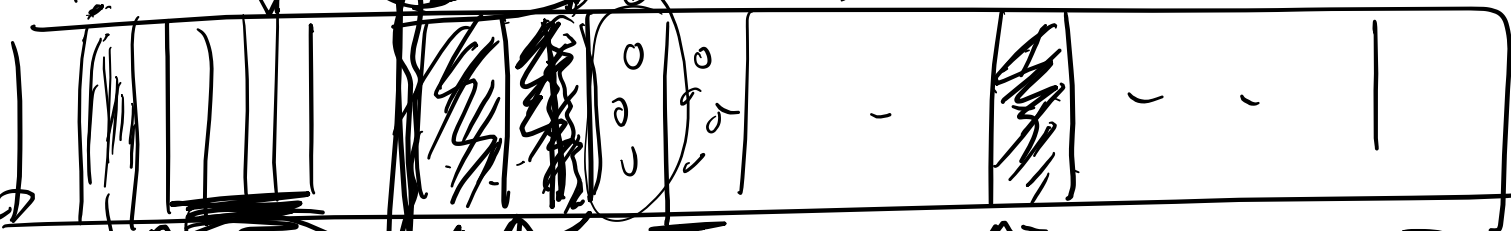
meta-data

↓ data blocks

ti-node #

1010101

unix.pptx



Super block

i-node array

hit map

512

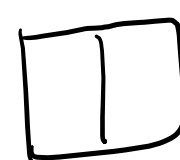
512

File



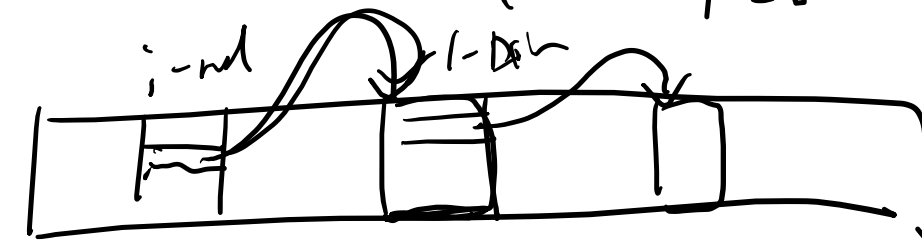
2K

4B



512B

128



i-node

i-dir

i-node

owner  
timestamp

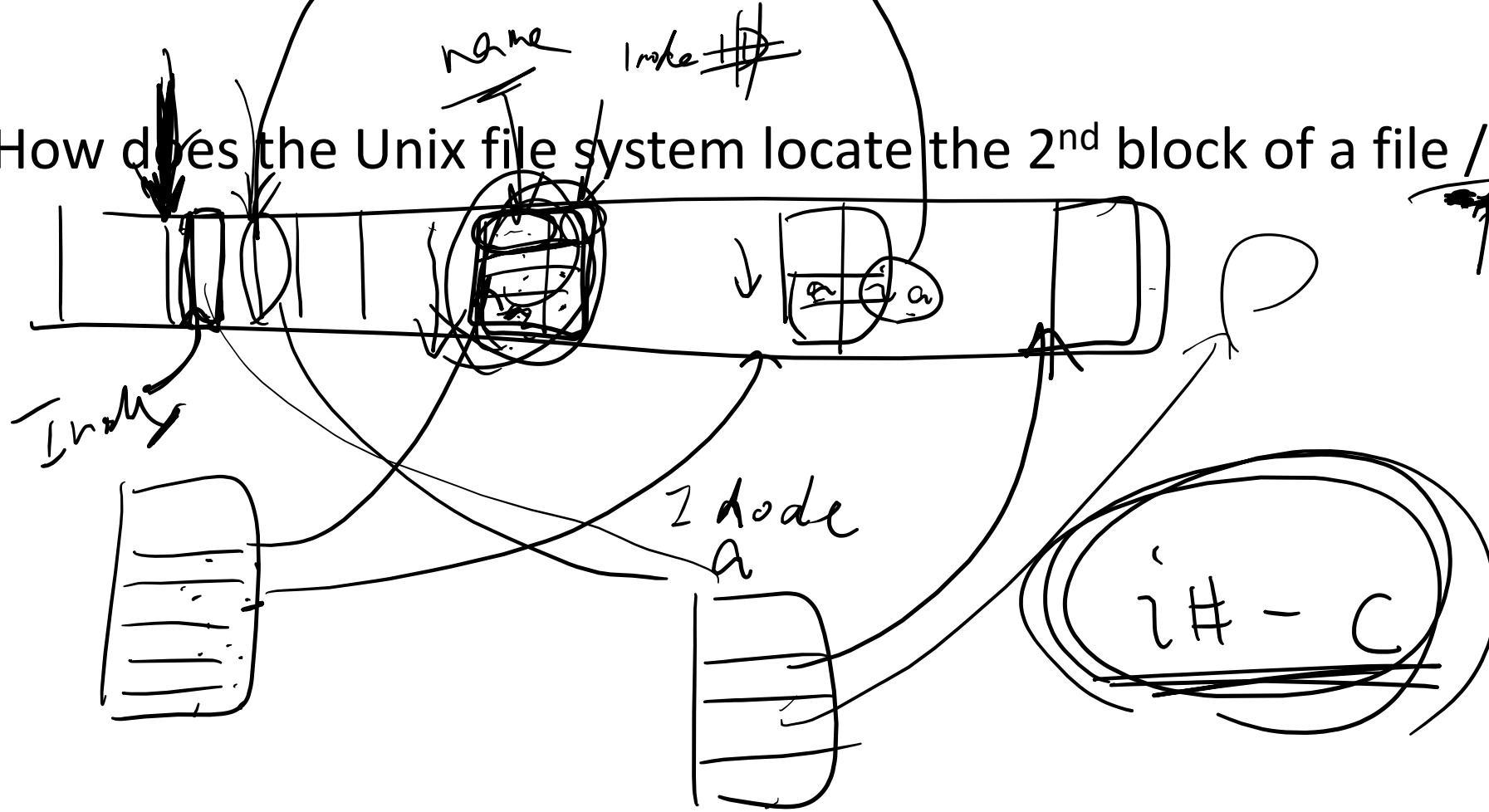
SB

8 entry

4K

# How to locate a data block in a file?

- How does the Unix file system locate the 2<sup>nd</sup> block of a file /a/b/c?

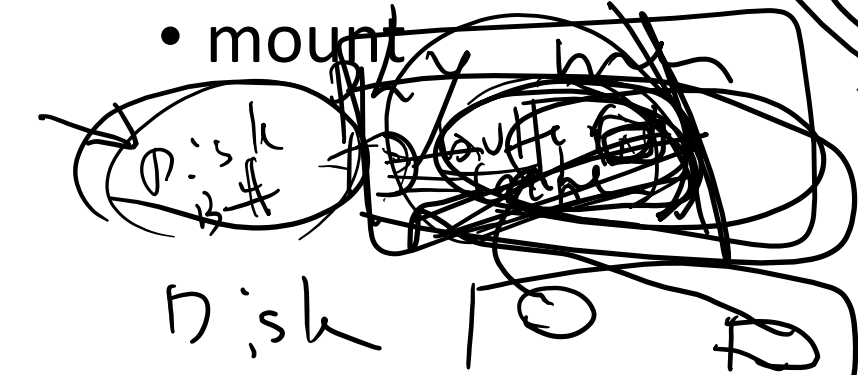


paging

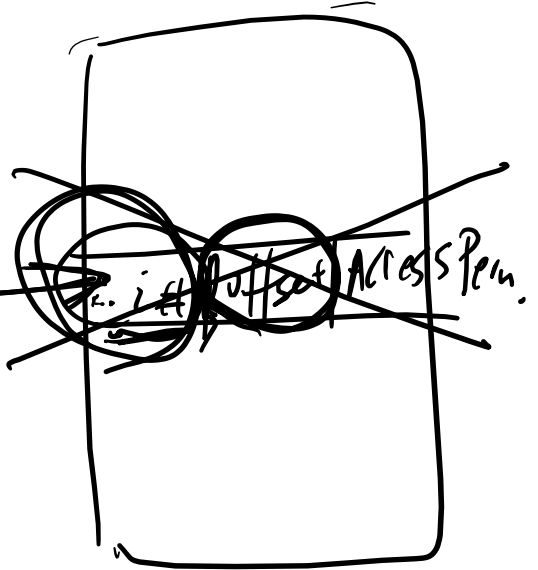
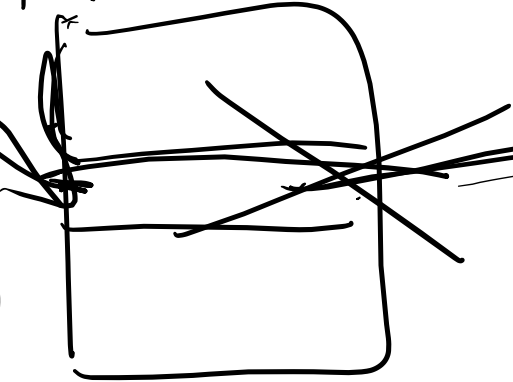
$\frac{p = \text{malloc}()}{*p \neq \text{---}}$

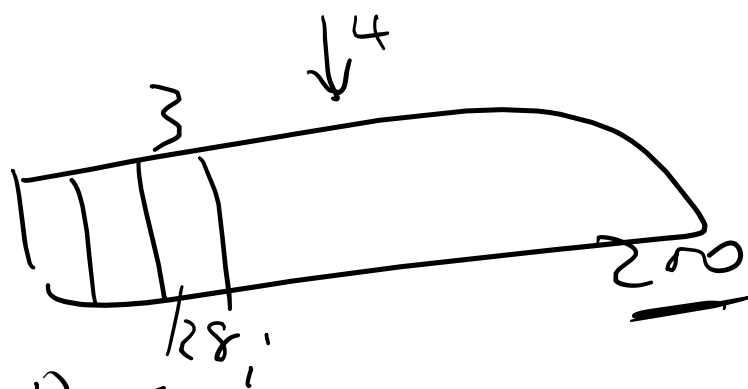
# File system system calls

- Open `fd = open("a/b/c", RD|WT)`
- Read `read(fd, buffer, 10);` ~~`read("/a/b/c", buffer, 10);`~~
- Write `write(fd, buffer, 10);` ~~`write("/a/b/c", buffer, 10);`~~
- Close
- link
- unlink
- mount



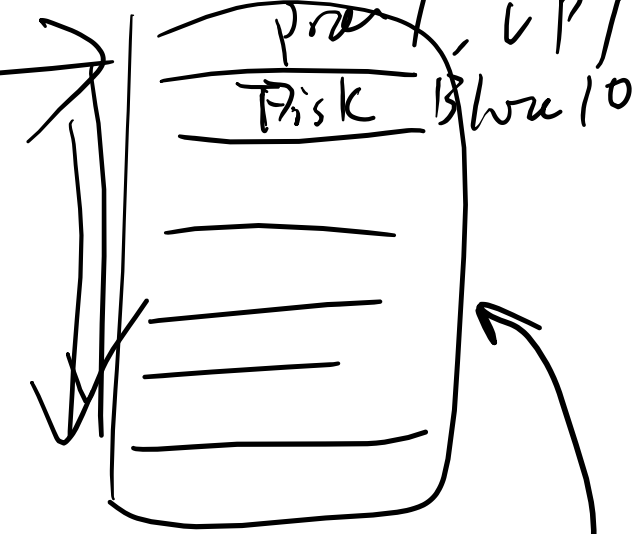
Process  
open file Table





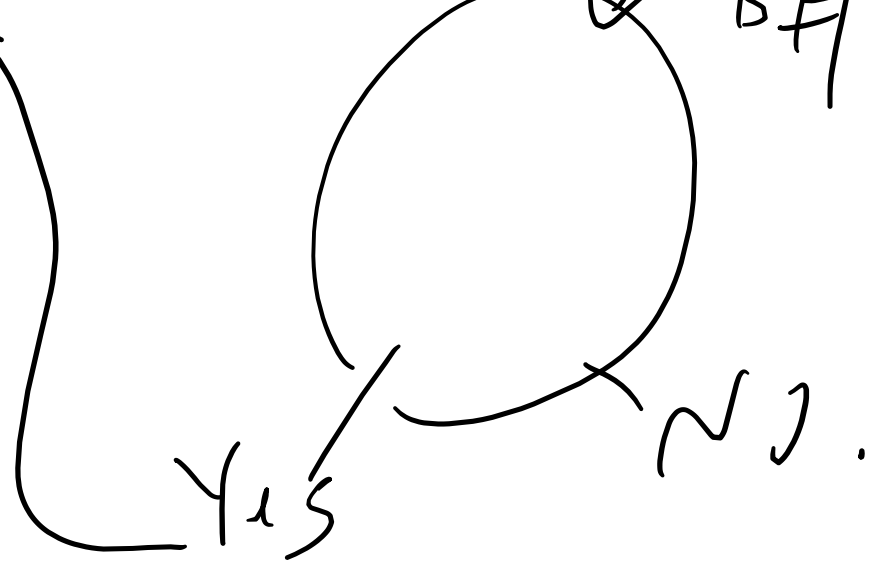
Phy Mem P  
Frame

Disk Block



Disk Block 4

Dr BH



# File protection and setuid

```
main () {  
  open(argv[1], WR);  
}
```

Alice

Other

~~tmp~~

s r w X

r-x

a.out (owner: Alice)

pwd

ls make  
/etc/passwd



# Performance optimization

- Many levels of buffering

/ Caching

# Processes and shell