## OS History




## OS History – Why study?

- Learn from the past
  - Why does an OS exist
  - Why things are the way they are
- Understand the issues
  - Many basic issues are the same
- Complexity is still an issue
- Some techniques come back

## OS History – Topics

- Vacuum tubes
- Batch systems
- Multiprogramming
- PCs




## History


Vacuum tube


Vacuum tube memory


ENIAC (1945 – 1955)


Deck of cards (problematic)


Punch card readers

Short history: http://ie.youtube.com/watch?v=Y-rXWWVmchY&NR=1
Part 2: http://ie.youtube.com/watch?v=6MlD_fEgoCc&feature=related
Short history: http://ie.youtube.com/watch?v=gas2Xi0rW6A&feature=related

## OS History

- Early computers
  - Each programme needed
    - to know about the h/w
    - device drivers for peripheral devices (e.g. printer)
  - … which was OK at the beginning but …
  - Programmes become more complex
    - ⇒Machines cam with **libraries** of support code which the programmer could use to make life easier (e.g. code for input and output)
    - ⇒Germ of early "OS"

## OS History

- Machines still one job at a time
  - e.g. Cambridge University washing line
  - Used different coloured pegs to indicate priority

## Batch Systems

- The monitor (an early "OS ") is the system s/w used to carry out instructions in a programme/job.
- Advantages
  - Move work to the computer operator
  - Increased performance (next job can start immediately after current job)
- Disadvantages
  - Long turn-around time from user viewpoint
  - Harder to debug programme
  - Lack of protection
    - one job could affect others
    - A job could corrupt the monitor
  - A job could enter an infinite loop (i.e. a job would never finish)

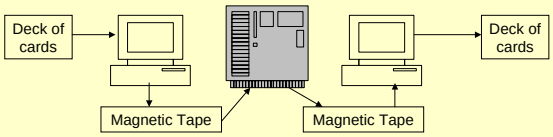
## Batch Systems

- Solutions
  - Simple memory protection scheme
  - 2 operating modes: monitor and user programmes
    - Only the monitor mode could perform I/O operations
  - Each job would be given a time limit in which it would hand control back to the monitor (to prevent infinite loops)

## Spooling Batch Systems
(mid 1960s – 1970s)

- Reading from magnetic tape was faster than deck of cards
  - ⇒ use slower computers to read a deck of cards onto tape
  - ⇒ Use faster computer for processing
- Use a timer so that jobs could be interrupted so that the monitor could do I/O operations
- I/O can now happen in parallel with computation

⇒ SPOOLing (Simultaneous Peripheral Operation OnLine)




## Multiprogramming
1960s -

- Can have several jobs in memory at once and each can be run for a certain period of time, in a given order
- The monitor is responsible for:
  - Starting jobs
  - Spooling
  - I/O
  - Switching between jobs
  - Protection
- Example:
  - 2 jobs: 1 I/O intensive (foreground), 1 CPU intensive (background)
  - CPU executes background job until foreground job needs some CPU time

## Multiprogramming
1960s -

- Programmers wanted to get around the batch process system
- Extended to allow multiple terminals to be connected to the computer
- OS uses **timesharing** used to **context-switch** between processes (i.e. determine which process has current access to the CPU)
- User-interactive process given higher **priority**
  - So that the user "appears" to have direct access
  - Done through **interrupts**
  - e.g when a key is pressed

## Personal Computers (PCs)
mid 1980s -

- Programmers could now afford their own computer
  - ⇒ Have complete access to it

… but still the need for multiprocessing (obviously)

- Now also:
  - Distributed systems (e.g. networking)
  - Mobile devices (e.g. mobile phone)
  - Wireless/wide area networking
  - Peer-to-peer systems
- H/W and S/W constantly changing and OS must adapt