

Computer Systems B

COMS20012

Introduction to Operating Systems and Security

A brief history of Operating Systems



Phase 0: no operating systems (1940-1955)

- Computer are exotic experimental equipments
- Program in machine language
- Use plugboard to direct computer
- Program manually loaded via card decks
- Goal: churn table of numbers (e.g. accounting)
- Progress:
 - People develop libraries that they share with each others
 - These libraries are the precursor to today OSs



Phase 1: Expensive computers, cheap people (1955-1970)

- Move the users away from the computer, give them (dumb) terminal
- OS is a batch monitors, a program that:
 - Load a user job
 - Run it
 - Move to the next
- Program fails? Record memory, save it, move on
- Efficient use of hardware (less “downtime”)
- Hard to debug



Phase 1: Technology

- Problems:
 - Utilization is low (one job at a time)
 - No protection between different jobs
 - Short jobs get stuck behind long one
- Solutions:
 - Hardware to the rescue: memory protection (separate code and data) and relocation (i.e. non-absolute addresses)
 - Multiprogramming: many users share the system
 - Scheduling: let short job finish quickly
 - OS must manage interactions between concurrent things (you learn that it is not easy last term)
 - OS becomes an important area of research
 - OS/360: first OS to run on a family of machine (big and small)



Phase 1: the bad

- Operating Systems did not really work
- OS/360 introduced in 1963... worked in 1968
- Systems were enormously complicated
- Written in assembly code
- No structured programming

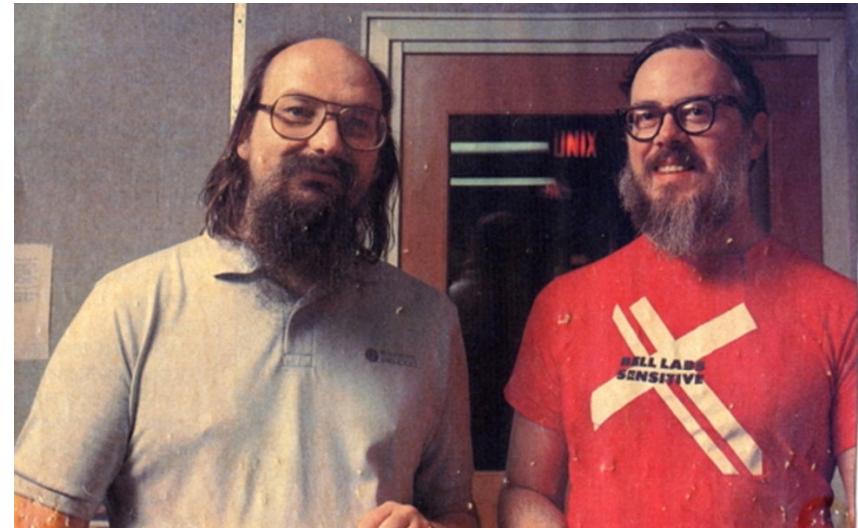
Phase 2: expensive computers, expensive people (1970-1980)

- Help people to be more productive
- Interactive timesharing: many people use the same machine at once
- Terminals are cheap: everyone get one! (e.g. Airline systems)
- Emergence of the file systems
- Try to give reasonable response time (avoid trashing)
- CTSS
 - Developed at MIT
 - One of the first time sharing system
 - Pioneered much of the work on scheduling
 - Motivation for MULTICS
- MULTICS
 - Joint development by MIT, Bell Labs, General Electric
 - One computer for everyone, people will buy computing as they buy electricity
 - Many seminal ideas: protected rings, hierarchical file systems, devices as files



Phase 2: UNIX

- Ken Thompson (former MULTICS' guy) wanted to use an old computer available at Bell Lab
- He and Dennis Ritchie built a system built by programmers for programmers
- Originally in assembly. Rewritten in C by Ritchie and Thompson
- New idea: portable OS (i.e. not tied to a specific hardware)
- Universities got the code for experimentation
- Berkeley added virtual memory support
- DARPA selected UNIX as its networking platform (ARPANET)
- UNIX becomes a commercial OS
- Important ideas popularized by UNIX
 - OS written in a high-level language (C)
 - OS is portable across hardware platforms
 - Pipes
 - Mountable file systems
 - Many, many more



Phase 3: cheap computer, expensive people (1980-1990)

- Put a computer in each terminal!
- CP/M first personal computer operating system.
- IBM needed an OS for its PC, CP/M being schedule
- Approached Bill Gates (Microsoft) to build one
- Gates approached Seattle Computer Products, bought 86-DOS, and created MS-DOS
- Goal: run CP/M programs and be ready quickly
- Market is broken horizontally
 - Hardware
 - OSes
 - Applications



Phase 4: Networked Systems (1990s-today)

- Connectivity is paramount!
- People want to share
- Networked applications propel industry
 - Email
 - Web
- Protection and multiprogramming less important for individual machines
- Protection and multiprogramming more important for servers
- New network-based architectures
 - Clusters
 - Grids
 - Distributed Operating Systems
 - Cloud
- Linux everywhere! (except in workstations)

Phase 5: the beginning of something new?

- Are we on the verge of something new?
- IoT, smartphones etc.

Thank you

bristol.ac.uk

