A Brief History of Operating Systems

- Phase 0: No operating system: 1940-1955
  - Computers are exotic experimental equipment.
  - Program in machine language.
  - Use plugboards to direct computer.
  - User sits at the console.
    - No overlap between computation, I/O, think time, and response time.
    - Programs manually loaded via card decks.
  - Goal: churn out tables of numbers.
  - Progress:
    - People developed libraries they could share with others.
    - Theses libraries were the precursor to today’s operating systems.

Phase 1: 1955-1970

- Computers are expensive; people are cheap
  - Make more efficient use of the computer: move the person away from the machine.
  - OS becomes a *batch monitor:* a program that loads a user's job, runs it, and then moves on to the next.
  - If program failed, the OS record the contents of memory and saves it somewhere.
  - More efficient use of hardware, but increasingly difficult to debug!

Phase 1 Technology

- Data channels and interrupts: allow overlap of I/O and computation.
  - Buffering and interrupt handling is done by OS.
  - Spool jobs onto “high speed” drums (cards are slow)
- Problems
  - Utilization is low (one job at a time).
  - No protection between different jobs.
  - Short jobs wait if they get stuck behind longer jobs.
- Solutions
  - Hardware to the rescue: memory protection and relocation
  - Multiprogramming: Many users can share the system.
  - Scheduling: Let short jobs finish quickly
  - OS must manage the interaction between concurrent things.
  - OS becomes an important science.
  - OS/360: first OS designed for a family of computers; one operating system designed to run from smallest to largest machines.

Phase 1 Disasters

- Operating systems didn’t really work!
- OS/360 was introduced in 1963; worked in 1968.
- Systems were enormously complicated.
- They were written in assembly code.
- No structured programming.
- Read Fred Brooks: *The Mythical Man Month*!

Phase 2: 1970-1980

- Computers and people are expensive
  - Help people be more productive.
  - Interactive timesharing: let many users use the same machine at once.
  - Terminals are cheap: give everyone one (e.g., Airline system)
  - Keep data on line: use fancy (and not so fancy) file systems.
  - Attempt to provide reasonable response time (avoid thrashing).
  - Marketplace is driven by vertical applications
- CTSS:
  - Developed at MIT.
  - One of the first timesharing systems.
  - Pioneered much of the work in scheduling.
  - Motivated MULTICS.
- MULTICS:
  - Joint development by MIT, Bell Labs, General Electric.
  - Envisioned one main computer to support “everyone”. People would buy computing services like electricity.
  - Many, many, many seminal ideas: protected rings, hierarchical file system, devices as files
  - Building it was more difficult than expected.
  - Technology caught up.

Phase 2: UNIX

- Ken Thompson (former Multician) wanted to use an old PDP-7 lying around Bell Labs.
- He and Dennis Ritchie built a system designed **by** programmers **for** programmers.
- Originally in assembly language. Rewritten by Ritchie and Thompson in C.
- New idea: portable operating system!
- Universities obtained code for experimentation.
- Berkeley added virtual memory support for the VAX.
- DARPA selected UNIX as its networking platform (arpanet).
- UNIX becomes a commercial operating system.
- Important ideas popularized by UNIX
    - OS written in a high-level language.
    - OS is portable across hardware platforms.
    - Pipes
    - Mountable file systems.
    - Many more (take 261 ...)

Phase 3: 1980-1990

- Computers are cheap; people are expensive.
  - Put a computer in each terminal!
  - CP/M first personal computer operating system.
  - IBM needed software for their PC’s, but CP/M was behind schedule.
  - Approached Bill Gates (Microsoft) to see if they could build one.
  - Gates approached Seattle Computer Products, bought 86-DOS, and created MS-DOS.
  - Primary goal: finish quickly and run existing CP/M programs.
  - OS becomes a subroutine library and command executive.

Phase 3 Technologies

- Personal workstations
  - The PERQ
  - The Xerox Alto
  - The SUN Workstation (Stanford University Network)
- Personal computers
  - The Apple II
  - The IBM PC
  - The Macintosh
- Business applications propel the industry
  - Word processors
  - Spreadsheets
  - Databases
- Marketplace is broken into horizontal markets
  - Hardware
  - Operating systems
  - Applications

Phase 4: Networked Systems (1990-200?)

- Connectivity is paramount.
- People want to share *data* not hardware.
- Networked applications propel the industry.
  - The Web
  - Email
- Protection and multiprogramming less important for personal machines.
- Protection and multiprogramming more important for server machines.
- Market continues horizontal stratification, add:
  - Internet service providers (service between OS and apps)
  - Information becomes a commodity.
  - Advertising becomes a computer marketplace.
- New network-based architectures:
  - Clusters
  - Grids
  - Distributed operating systems
  - Cloud (or is this a new generation?)

Phase 5: ???

- Are we on the verge of another generation?
- If so, what is it?