Chapter 1
Operating Systems

User View
- Ease of Use, i.e., convenience
- Resource Utilization, i.e., system efficiency
  - Personal Computer (connected to the internet)
  - Terminal (dumb) connected to a Mainframe/Minicomputer
  - Workstation connected to a Server
  - Supercomputer

System View
- Resource Allocator, i.e., Government providing an Environment which supports a fair allocation of the available resources to the competing application programs
- Control Program

Operating System
- Kernel
- Integral Support Programs
  - memory management
  - process control
  - etc
- Utilities
  - Housekeeping Services
    - Make directories
    - Delete directories
    - Delete files
    - Move directories
    - Move files
    - etc
  - Maintenance Services
    - Backup
    - Virus Detection
    - Encryption
    - etc
- Shells
  - Bourne Shell
  - Korn Shell
  - C Shell
  - Bash Shell

Operating System Management Functions
- Processes
- Memory
- Storage
- I/O Allocation
- User Interfaces
- Security
- Protection
- Communication
- Errors
- Accounting
Operation
ROM/EPPROM/BIOS – Firmware -- bootstrap program
checks system, loads O/S kernel, initiates kernel execution

Interrupts (redirects CPU activity)
- Hardware – device activation, e.g., keyboard, memory, etc.
- Software – system call

Interrupt
- Stops CPU
- Stores process information in Process Control Block (PCB)
- Transfers control to a fixed location which contains the
  service routine required for processing the specific interrupt
- Additional interrupts are held in a service queue
  until the current interrupt is processed
- Once the interrupt is processed, control returns to the interrupted program
- Interrupts held in the service queue are processed

Interrupt Request
with Device Number 3

```
Device 3 Service Routine
...
...
... return
```

```
<table>
<thead>
<tr>
<th>Interrupt Vector</th>
</tr>
</thead>
<tbody>
<tr>
<td>Device 1 Service Routine Address</td>
</tr>
<tr>
<td>Device 2 Service Routine Address</td>
</tr>
<tr>
<td>Device 3 Service Routine Address</td>
</tr>
<tr>
<td>Device 4 Service Routine Address</td>
</tr>
<tr>
<td>Device 5 Service Routine Address</td>
</tr>
<tr>
<td>etc</td>
</tr>
</tbody>
</table>
```

```
Process1 Control Block
Return Address
Program Counter Contents
Instruction Register Contents
Program Status Word
...
```

```
Process2 Control Block

Process3 Control Block
```

Stack
Von Neumann Architecture

Memory – volatile
Semiconductor Memory
   DRAM (dynamic random access memory)

USB Flash Drives
   http://en.wikipedia.org/wiki/USB_flash_drive

Storage – nonvolatile
Magnetic or Laser-based Media

Device Controllers
- Local Buffer (Storage Space)
- Special Purpose Registers
- Attached Peripheral Devices
- Moves data between Local Buffer & selected Peripheral Device

Operating System
   Memory System

   Device Driver

   Device Controller

   Peripheral Device

Small Data Transfers
O/S I/O Request (one byte) ® Device Driver ® Loads DC Registers ® DC starts Device ®
Device informs DC of action completion ® DC interrupts the Device Driver with Status ®
Device Driver returns control to O/S with status information

Large Data Transfers
O/S Specifies Data Transfer Bulk (beginning address, number of data bytes) ® DMA DC
DMA DC ® downloads entire block of data directly from memory w/o CPU intervention
DMA DC feeds Device Driver until project is complete ® interrupts CPU

Multiprocessor Systems
- Parallel Systems
- Tightly Coupled Systems
  - two or more processors (on separate chips) sharing the system bus
  - possibly sharing the clock, memory &/or peripheral devices

Advantages
   Graceful Degradation
   Fault Tolerant
   HP NonStop
      multiple pairs of CPU's
      lock step execution
voting; invoking an additional pair of CPU's to re-execute the instruction

Asymmetric Multiprocessing (ASMP)
Master Processor
- controls system
- schedules and allocates work
- assigns tasks to slave processors
- some slave processors may have predefined tasks

Symmetric Multiprocessing (SMP)
Any processor may perform any task required by the O/S, i.e.,
the processors are all peers

page 15 figure 1.6

Solaris, Windows XP, Mac OS/X, Linux

Multiple Core Systems
- two or more CPU's on a single chip
- on-chip communication is faster than communication between chips
- one chip with multiple cores uses significantly less power than multiple chips

Server Systems
- Database Servers
- Web Servers

Blade Servers
- Multiple Processor Boards
- I/O Boards
- Networking Boards

Each Processor Board boots on its own and runs it own O/S
Blade Server provides multiple independent multiprocessing systems

Clustered Systems
- Independent Systems
- Shared Storage
- Closely Linked
  - Internet
  - High Speed Interconnect

Asymmetric Clustering
- hot standby machine monitors active server;
- if active server fails, hot standby machine becomes the active server

Symmetric Clustering
- Two or more hosts acting as active servers and monitoring one another; if one of the active servers fail, the other machines assume its duties

High-Availability Service
- Cluster Software
- each node monitors its neighbors
if a monitored neighbor fails, its workload is assumed by its neighbors

High-Performance Computing
Producing Computer Programs, i.e., program code that will take advantage of multiple systems, i.e., parallelization

Parallel Clusters
Multiple Hosts accessing the same data on shared storage
Access Control & Data Locking
  Oracle Real Application Cluster

Operating System Structure

Multiprogramming
  Efficiency
  • CPU
  • I/O Devices

  Job Pool – Multiple Jobs -- disk resident
  Multiple Jobs – memory resident

  O/S picks job to execute
  I/O requests results in removing job from execution
  O/S picks jobs from Job Pool to place in Memory
  O/S picks jobs to move from memory to Job Pool (swapping)

Timesharing or Multitasking
  switches jobs every few milliseconds
  provides an interactive environment
  response time is critical

  job scheduling
  CPU scheduling
  virtual memory
  physical memory vs. logical memory
O/S Operations

Operating Systems are interrupt, i.e., event, driven
- Exceptions are software interrupts generated by errors, e.g., divide by zero
- Traps are software interrupts generated by software requests, e.g., I/O request
- Hardware Interrupts are generated by hardware actions, e.g., keyboard interrupt &/or hardware failures, e.g., memory failure

Interrupt Service Routine
- Routine provided to service a particular type of interrupt

Dual-Mode Operation
- User Mode
  - Kernel Mode, Supervisory Mode, System Mode, Privileged Mode

  Instruction Word contains a Mode Bit
  - Mode Bit indicates the current mode
    - 0 ➔ kernel
    - 1 ➔ user
  - System Call: Mode Bit 1 ➔ 0
  - System Call Return: Mode Bit 0 ➔ 1

  Changing the Mode Bit is a Privileged Instruction

Boot Time – hardware starts in kernel mode
Operating System starts all user applications in user mode
Whenever a Hardware Interrupt, Trap, or Exception occurs, the hardware switches from user mode to kernel mode
The Operating System always switches to user mode before passing control to a user program

Privileged Instructions
- machine instructions that could cause harm can be executed in kernel mode
- if execution in user mode is attempted, it is trapped to the operating system as an illegal instruction
  - Mode Bit Modification Instruction
  - I/O Control Instructions
  - Time Management
  - Interrupt Management

System Call Execution – Silberschatz page 22 paragraph 6, 7
Silberschatz page 23 paragraph 1

READ!!
Timer

CPU Control
- Infinite Loops
- Not Relinquishing the CPU, e.g., not Requesting I/O Services

O/S sets the timer with the allotted time; sets timer to interrupt; turns control over to user

When the timer interrupts, the control is returned to the O/S

Prevents a process from running too long.

Process Management

Process – active entity
- program in execution
- resources
  - CPU Time Allotment
  - Memory Allocation
  - Files
  - I/O Devices
  - Input (Initial Data)

Single Threaded Process – Single Program Counter

Multiple Threaded Process – Multiple Program Counters

Each Program Counter pointing to the next instruction to execute for the associated thread

A process is a unit of work.
System consists of a collection of processes.
- Operating System Processes
- User Processes
All may potentially execute concurrently, e.g., multiplexing on a single CPU

Silberschatz -- page 24

Memory Management
Silberschatz -- pages 24-25

Storage Management
Silberschatz -- pages 25-29

File Management System
Logical Storage Unit – File

Mass-Storage Management
Secondary Storage – magnetic disk
Tertiary Storage – magnetic tapes, dvd’s

Caching

Hardware Caches

Data Stream

Instruction Stream

Internal Programmable Registers, e.g., A, X, R₀, ..., R₁₅, Q, etc.
L₁ – closest to CPU; may be inside the chip
L₂ – next closest; on the chip but not inside or next to the chip
L₃ – in the vicinity of the chip

http://en.wikipedia.org/wiki/Cache_Memory#Multi-level_caches

cache management

<table>
<thead>
<tr>
<th>cache size</th>
<th>replacement policy</th>
</tr>
</thead>
</table>

Silberschatz page 27 figure1.11

movement control

| hardware control: | memory ←→ cache ←→ registers |
| O/S control:      | memory ←→ disk               |

data integrity

A selected data item A may appear in several different levels of storage, i.e., disk, memory, L₃ cache, L₂ cache, L₁ cache, registers

I/O request → copy disk block to memory buffer area → copy segment of buffer area to L₃ cache → copy smaller section of L₃ cache to L₂ cache → copy yet a smaller section of L₂ cache to L₁ cache → copy word from L₁ cache to selected register

The selected data item will, in all probability, have different values in various parts of the storage hierarchy!

In multitasking system care must be taken to ensure that all threads are computing with the most recent value of the data item.

cache coherency

In a multiprocessor system where each CPU maintains a local cache as well as internal registers, an update in a specific cache must be reflected immediately in all the other caches in which the data item resides
I/O Systems
  Memory Management
  • Buffering
  • Caching
  • Spooling
  File System – Device Controller Interface
  Device Drivers for Specific hardware devices

Protection & Security
Protection – controlling access of users or processes to system resources
  • Detecting latent errors at the interface of subsystems
  • Unauthorized &/or incompetent users
Security – detect, defend against & defeat internal & external attacks
  • Worms
  • Viruses
  • Denial of Service
  • Identity Theft
  • Service Theft

Enforcement Options
  • Operating System
  • Policy
  • Software

User Identification
  • UID – Unix, et.al.
  • SID – Windows

Group Identification

Privilege Escalation – Effective UID

Real-Time Systems
  • general purpose computers – special purpose systems
  • Embedded Systems
    o special purpose processors – o/s tailored to specific requirements
    o processors with application specific circuits – ASIC
  • well-defined, fixed time constraint, i.e., rigid time requirements
  • control device for dedicated application
  • input – sensors
  • output – adjust controls
Multimedia Systems
  Audio
  Video
  Bandwidth

Handheld Systems
  size limitations
  processor speed
  i/o limitations, e.g., keyboards, displays, etc.

Computing Environments
  Legacy Systems
    • Batch Systems
    • Time Sharing Systems
  Client-Server Systems
  File-Server Systems
    • Web-Server Systems
  Peer-to-Peer Systems (P2P)
    • Node joins a P2P network
      o Node registers it’s services with a central lookup service on the network
      o Network provides a discovery service that nodes can use to locate a specific service
    • Nodes can act as both clients and as servers
  Web-based Systems
    • Applications reside on the web, provided free or for a fee

Open-Source O/S
  Silberschatz pages 37-42
  Linux (Torvald)
  Sun
  GNU
  GNU General Public License
  Free Software Foundation (FSF)

Closed-Source O/S
  Microsoft
  IBM
  DEC
  US Digital Millennium Copyright Act

Linux Distributions
  • Redhat – commercial use
  • SUSE
  • Fedora
  • Debian
  • Slackware
  • Ubuntu – vmware player access
    Silberschatz page 39

Solaris
  • Originally based on BSD Unix
  • Some code is still owned by AT&T and other companies
  • Can be compiled and linked to restricted binaries
  • Silberschatz page 40