Embedded Operating System

Shell Script

- **Shell Scripts**
  - Shell scripts are interpreted by bash shell (not compiled).
  - Interpreter -- Line by line, hence it is slow.

  - Shell scripts are simpler syntax than programming languages.
  - No pointers, No recursion, No structure, ...

  - **Applications**
    - Administrators use shell scripts for routine tasks.
    - Creating multiple users in a series.
    - Executing set of commands in a sequence or with some conditions.
    - Product installation or maintenance.

Classification of OS

- OS can be categorized based on the target system (computers).
  - Mainframe systems
  - Desktop systems
  - Multi-processor (Parallel) systems
  - Distributed systems
  - Hand-held systems
  - Real-time systems

Mainframe systems

- Examples: UNIX and its flavours, IBM-360, etc.
- **Resident Monitor**
  - Early (oldest) OS resides in memory and monitor execution of the programs. If it fails, error is reported.
  - OS provides hardware interfacing that can be reused by all the programs.

- **Batch Systems**
  - The batch/group of similar programs is loaded in the computer, from which OS loads one program in the memory and execute it.
  - The programs are executed one after another.
  - In this case, if any process is performing IO, CPU will wait for that process and hence not utilized efficiently.

- **Multi-Programming**
  - In multi-programming systems, multiple program can be loaded in the memory.
  - The number of program that can be loaded in the memory at the same time, is called as "degree of multi-programming".
  - In these systems, if one of the process is performing IO, CPU can continue execution of another program.
  - This will increase CPU utilization.
  - Each process will spend some time for CPU computation (CPU burst) and some time for IO (IO burst).
  - If CPU burst > IO burst, then process is called as "CPU bound".
  - If IO burst > CPU burst, then process is called as "IO bound".
  - To efficiently utilize CPU, a good mix of CPU bound and IO bound processes should be loaded into memory.
  - This task is performed by an unit of OS called as "Job scheduler" OR "Long term scheduler".
  - If multiple programs are loaded into the RAM by job scheduler, then one of process need to be executed (dispatched) on the CPU.
  - This selection is done by another unit of OS called as "CPU scheduler" OR "Short term scheduler".

- **Multi-tasking OR time-sharing**
  - CPU time is shared among multiple processes in the main memory is called as "multi-tasking".
  - In such system, a small amount of CPU time is given to each process repeatedly, so that response time for any process < 1 sec.
  - With this mechanism, multiple tasks (ready for execution) can execute concurrently.
  - There are two types of multi-tasking:
    - **Process based multitasking**:
      - Multiple independent processes are executing concurrently.
      - Processes running on multiple processors called as "multi-processing".
    - **Thread based multi-tasking OR multi-threading:**
      - Multiple parts/functions in a process are executing concurrently.

- **Multi-user**
  - Multiple users can execute multiple tasks concurrently on the same systems.
  - e.g. IBM 360, UNIX, Windows Servers, etc.
  - Each user can access system via different terminal.
  - There are many UNIX commands to track users and terminals.
    - tty (teletype) : It prints the name of the current terminal
    - who : Information about currently logged in users, system boot time, run level,processes,…
    - who am i : Gives you the name of the current user, the terminal they are logged in at, the date and time when they logged in.
    - whoami : It gives username of the current user
    - w : Displays the users.

Desktop systems

- Personal computers -- desktop and laptops
- User convenience and Responsiveness
- Examples: Windows, Mac, Linux, few UNIX, ...

Multiprocessor systems

- The systems in which multiple processors are connected in a close circuit is called as "multiprocessor computer".
- The programs/OS take advantage of multiple processors in the computer are called as "Multi-procssing" programs/OS.
  - Windows Vista: First Windows OS designed for multi-processing.
  - Linux 2.5+ : Linux started supporting multi-processing.
  - terminal> uname -a
- Since multiple tasks can be exeuted on these processors simultaneously, such systems are also called as "parallel systems".
- Parallel systems have more throughput (Number of tasks done in unit time).
- There are two types of multiprocessor systems:
  - Asymmetric Multi-processing
  - Symmetric Multi-processing

- **Asymmetric Multi-processing**
  - OS treats one of the processor as master processor and schedule task for it.
  - The task is in turn divided into smaller tasks and get them done from other processors.

- **Symmetric Multi-processing**
  - OS considers all processors at same level and schedule tasks on each processor individually.
  - All modern desktop systems are SMP.

- **Distributed systems**
  - Multiple computers connected together in a close network is called as "distributed system".
  - Its advantages are high availability (24x7), high scalability (many clients, huge data), fault tolerance (any computer may fail).
  - The requests are redirected to the computer having less load using "load balancing" techniques.
  - The set of computers connected together for a certain task is called as "cluster".
  - Examples: Linux

- **Handheld systems**
  - OS installed on handheld devices like mobiles, PDAs, iPODs, etc.
  - Challenges:
    - Small screen size
    - Low end processors
    - Less RAM size
    - Battery powered
  - Examples: Symbian, iOS, Linux, PalmOS, WindowsCE, etc.

- **Realtime systems**
  - The OS in which accuracy of results depends on accuracy of the computation as well as time duration in which results are produced, is called as "RTOS".
  - If results are not produced within certain time (deadline), catastrophic effects may occur.
  - These OS ensure that tasks will be completed in a definite time duration.
  - Time from the arrival of interrupt till begin handling of the interrupt is called as "Interrupt Latency".
  - RTOS have very small and fixed interrupt latencies.
  - RTOS Examples: uC-OS, VxWorks, pSOS, RTLinux, FreeRTOS, etc

Thank you!

Kiran Jaybhave

email – kiran.jaybhave@sunbeaminfo.com