Chapter 3 Overview of Operating Systems

- Computing Environments and Nature of Computations
- Classes of Operating Systems
- Efficiency, System Performance, and User Service
- Batch Processing Systems
- Multiprogramming Systems
- Time-Sharing Systems
- Real-Time Operating Systems
- Modern Operating Systems

Computing Environments and Nature of Computations

- A computing environment consists of a computer system, its interfaces with other systems, and the services provided by its operating system to its users and their programs
- Evolution:
  - Noninteractive Computing Environments
  - Interactive Computing Environments
  - Real-Time, Distributed, and Embedded Environments
  - Modern Computing Environments
Computing Environments and Nature of Computations (continued)

- Noninteractive Computing Environments
  - User submits program and data to OS; user does not interact with computation
  - Computations in form of program or job
  - OS focuses on efficient use of resources
- Interactive Computing Environments
  - User may interact with computation
  - OS deals with execution of programs, not jobs
    - Execution of a program is called a process
    - User interacts with a process by presenting subrequests
  - OS focuses on providing quick response to subrequests made to processes

---

Table 3.1 Computations in an OS

<table>
<thead>
<tr>
<th>Computation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Program</td>
<td>A program is a set of functions or modules, including some functions or modules obtained from libraries.</td>
</tr>
<tr>
<td>Job</td>
<td>A job is a sequence of programs that together achieve a common goal. It is not meaningful to execute a program in a job unless previous programs in the job have been executed successfully.</td>
</tr>
<tr>
<td>Process</td>
<td>A process is an execution of a program.</td>
</tr>
<tr>
<td>Subrequest</td>
<td>A subrequest is the presentation of a computational requirement by a user to a process. Each subrequest produces a single response, which consists of a set of results or actions.</td>
</tr>
</tbody>
</table>
Computing Environments and Nature of Computations (continued)

• A real-time computation has specific time constraints
  – OS in real-time environment ensures computations complete within time constraints

• Distributed computing environment: enables a computation to use resources located in several computer systems through a network

• Embedded computing environment: computer system is a part of a specific hardware system and runs computations that control the system
  – OS has to meet the time constraints arising from the nature of the system being controlled

Computing Environments and Nature of Computations (continued)

• Modern Computing Environments
  – Has features of several of the computing environments described earlier
  – OS uses complex strategies to manage user computations and resources
Classes of Operating Systems

Table 3.2  Key Features of Classes of Operating Systems

<table>
<thead>
<tr>
<th>OS class</th>
<th>Period</th>
<th>Prime concern</th>
<th>Key concepts</th>
</tr>
</thead>
<tbody>
<tr>
<td>Batch processing</td>
<td>1960s</td>
<td>CPU idle time</td>
<td>Automate transition between jobs</td>
</tr>
<tr>
<td>Multiprogramming</td>
<td>1960s</td>
<td>Resource utilization</td>
<td>Program priorities, preemption</td>
</tr>
<tr>
<td>Time-sharing</td>
<td>1970s</td>
<td>Good response time</td>
<td>Time slice, round-robin scheduling</td>
</tr>
<tr>
<td>Real time</td>
<td>1980s</td>
<td>Meeting time constraints</td>
<td>Real-time scheduling</td>
</tr>
<tr>
<td>Distributed</td>
<td>1990s</td>
<td>Resource sharing</td>
<td>Distributed control, transparency</td>
</tr>
</tbody>
</table>

Efficiency, System Performance, and User Service

• Two of the fundamental goals of an OS:
  – Efficiency of use
    • Efficiency of a resource = percentage of total resource used by user processes
  – User convenience
    • Measurable aspect: User service
      – Turnaround time
      – Response time

• To a system administrator, system performance is more important
  – Typically measured as throughput
Efficiency, System Performance, and User Service (continued)

Table 3.3  Measures of Efficiency, System Performance, and User Service

<table>
<thead>
<tr>
<th>Aspect</th>
<th>Measure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Efficiency of use</td>
<td>CPU efficiency</td>
<td>Percent utilization of the CPU</td>
</tr>
<tr>
<td></td>
<td>Memory efficiency</td>
<td>Percent utilization of memory</td>
</tr>
<tr>
<td>System performance</td>
<td>Throughput</td>
<td>Amount of work done per unit time</td>
</tr>
<tr>
<td>User service</td>
<td>Turnaround time</td>
<td>Time to complete a job or a process</td>
</tr>
<tr>
<td></td>
<td>Response time</td>
<td>Time to implement one subrequest</td>
</tr>
</tbody>
</table>

Definition 3.1 Throughput: The average number of jobs, programs, processes, or subrequests completed by a system in unit time.

Definition 3.2 Turnaround Time: The time from submission of a job, program, or process by a user to the time its results become available to the user.

Definition 3.3 Response Time: The time from submission of a subrequest by a user to the time a process responds to it.

3.9

Batch Processing Systems

- Used in noninteractive computing environments
- Batch: sequence of user jobs formed for processing by the OS
  - One-job-at-a-time
  - One-program-at-a-time within a job
- Batching kernel initiates processing of jobs without requiring computer operator’s intervention
- Control statements used to protect against interference between jobs
  - Command interpreter read a card when currently executing program in job wanted the next card
  - Abort the program if the program tried to read more data than provided

3.10
// JOB — "Start of job" statement
// EXEC FORTRAN → Execute the Fortran compiler

	Fortran program

// EXEC → Execute just compiled program

	Data for Fortran program

/* → "End of data" statement

*/ → "End of job" statement

Figure 3.2 Control statements in IBM 360/370 systems.