Event Handling

• Events that occur during the operation of an OS:
  1. Process creation event
  2. Process termination event
  3. Timer event
  4. Resource request event
  5. Resource release event
  6. I/O initiation request event
  7. I/O completion event
  8. Message send event
  9. Message receive event
  10. Signal send event
  11. Signal receive event
  12. A program interrupt
  13. A hardware malfunction event

Event Handling (continued)

• When an event occurs, the kernel must find the process whose state is affected by it
  – OSs use various schemes to speed this up
    • E.g., event control blocks (ECBs)

Figure 5.7 Event control block (ECB).

Figure 5.8 PCB-ECB interrelationship.
Sharing, Communication, and Synchronization Between Processes

Table 5.7  Four Kinds of Process Interaction

<table>
<thead>
<tr>
<th>Kind of interaction</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data sharing</td>
<td>Shared data may become inconsistent if several processes modify the data at the same time. Hence processes must interact to decide when it is safe for a process to modify or use shared data.</td>
</tr>
<tr>
<td>Message passing</td>
<td>Processes exchange information by sending messages to one another.</td>
</tr>
<tr>
<td>Synchronization</td>
<td>To fulfill a common goal, processes must coordinate their activities and perform their actions in a desired order.</td>
</tr>
<tr>
<td>Signals</td>
<td>A signal is used to convey occurrence of an exceptional situation to a process.</td>
</tr>
</tbody>
</table>