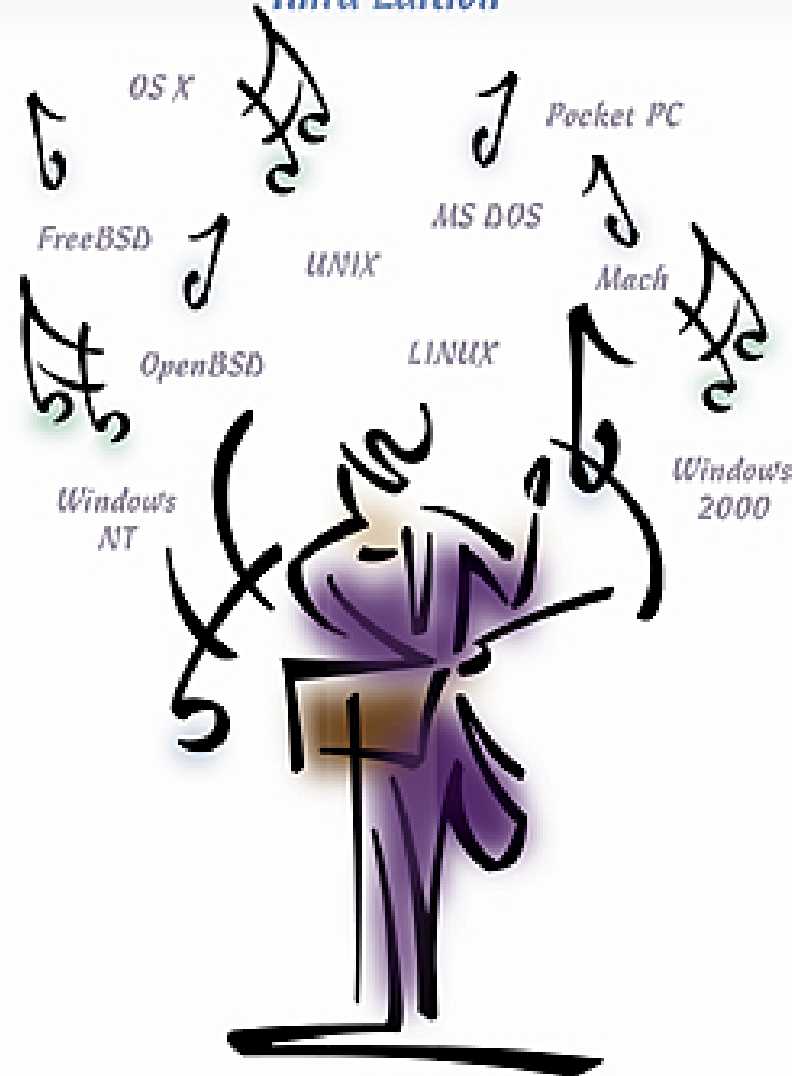


OPERATING SYSTEMS

Third Edition

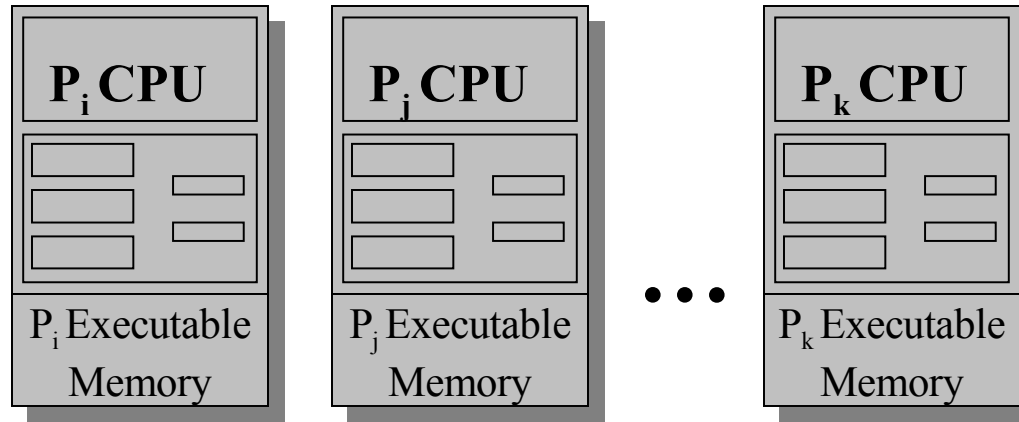


GARY NUTT

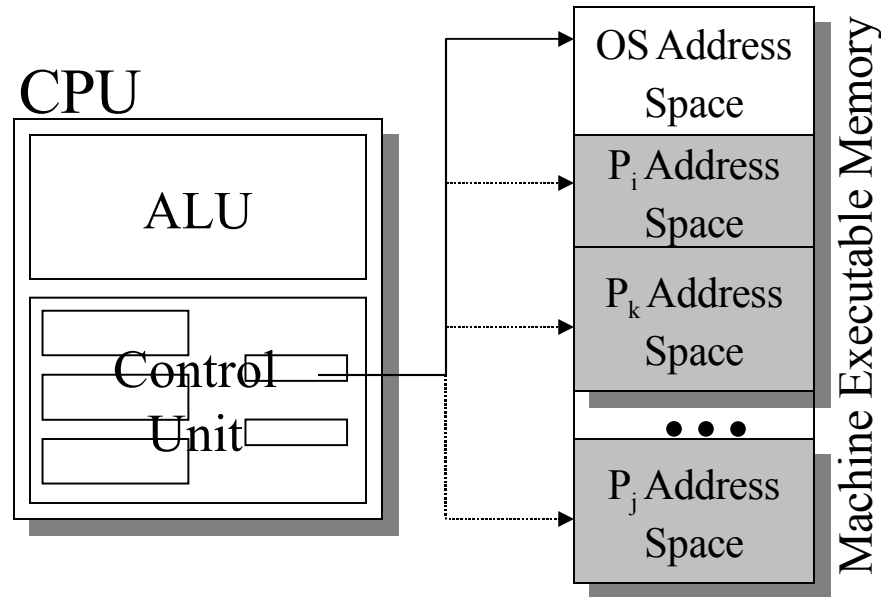
Implementing Processes, Threads, and Resources



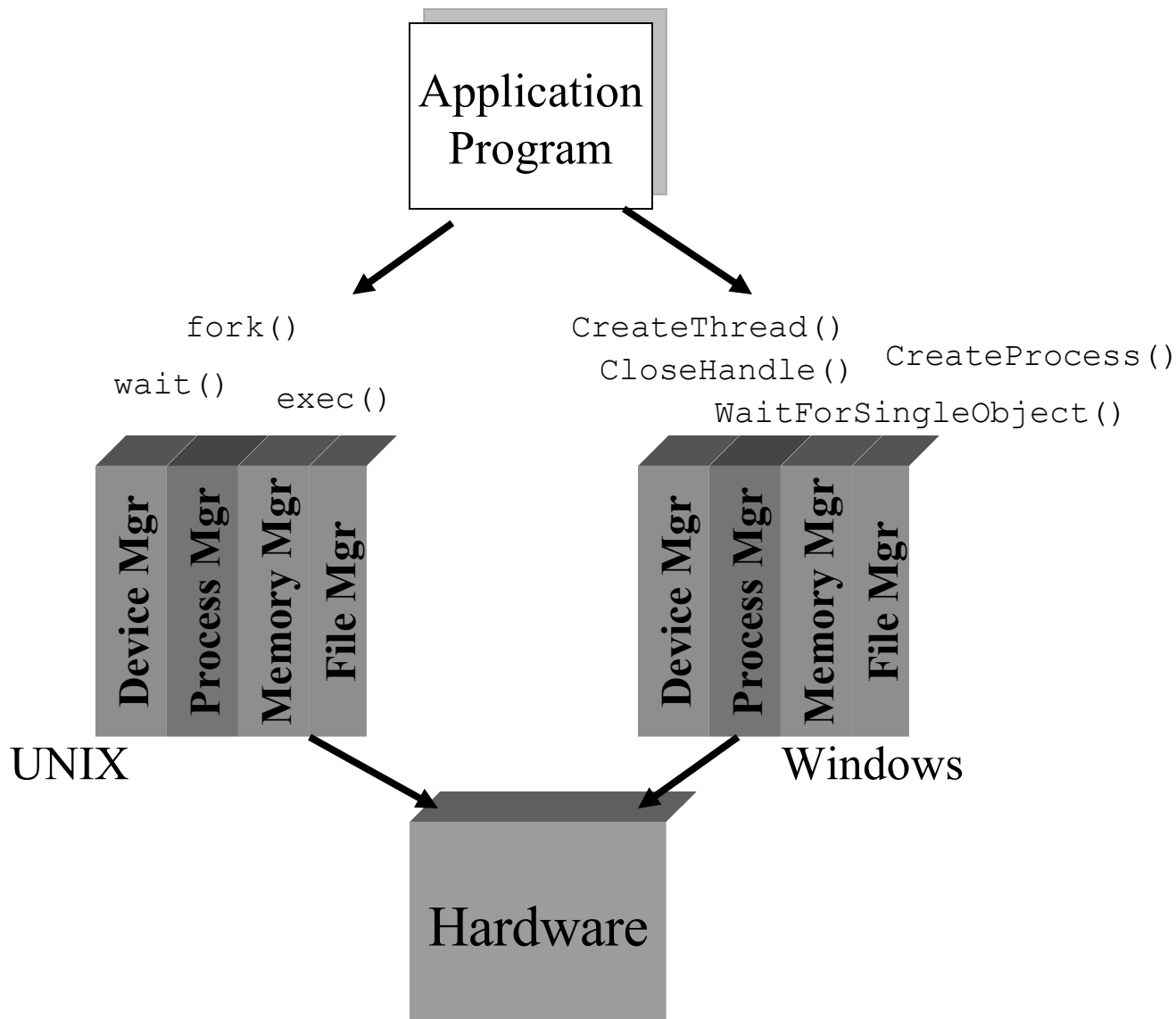
Implementing the Process Abstraction



OS interface



External View of the Process Manager

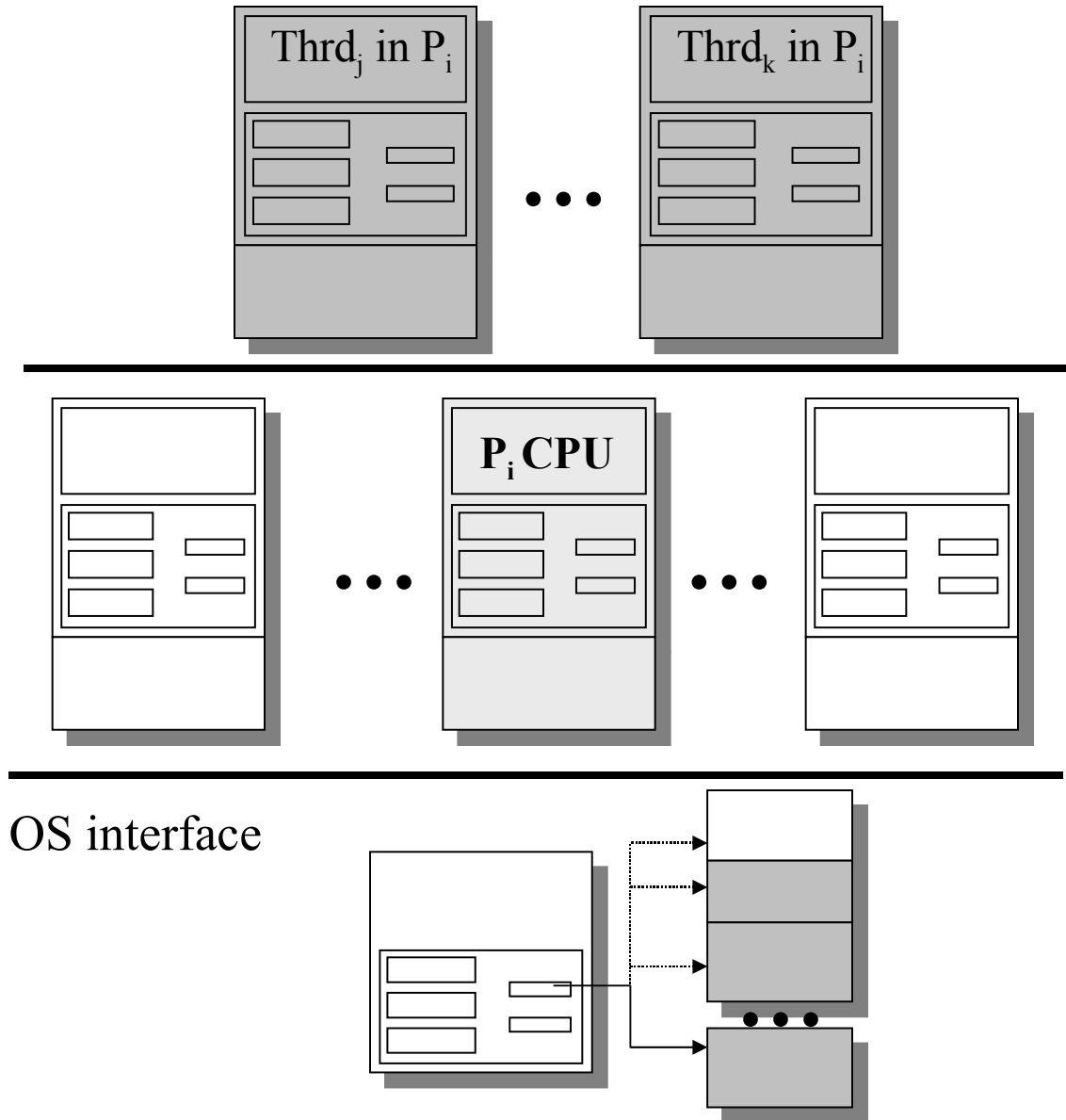


Process Manager Responsibilities

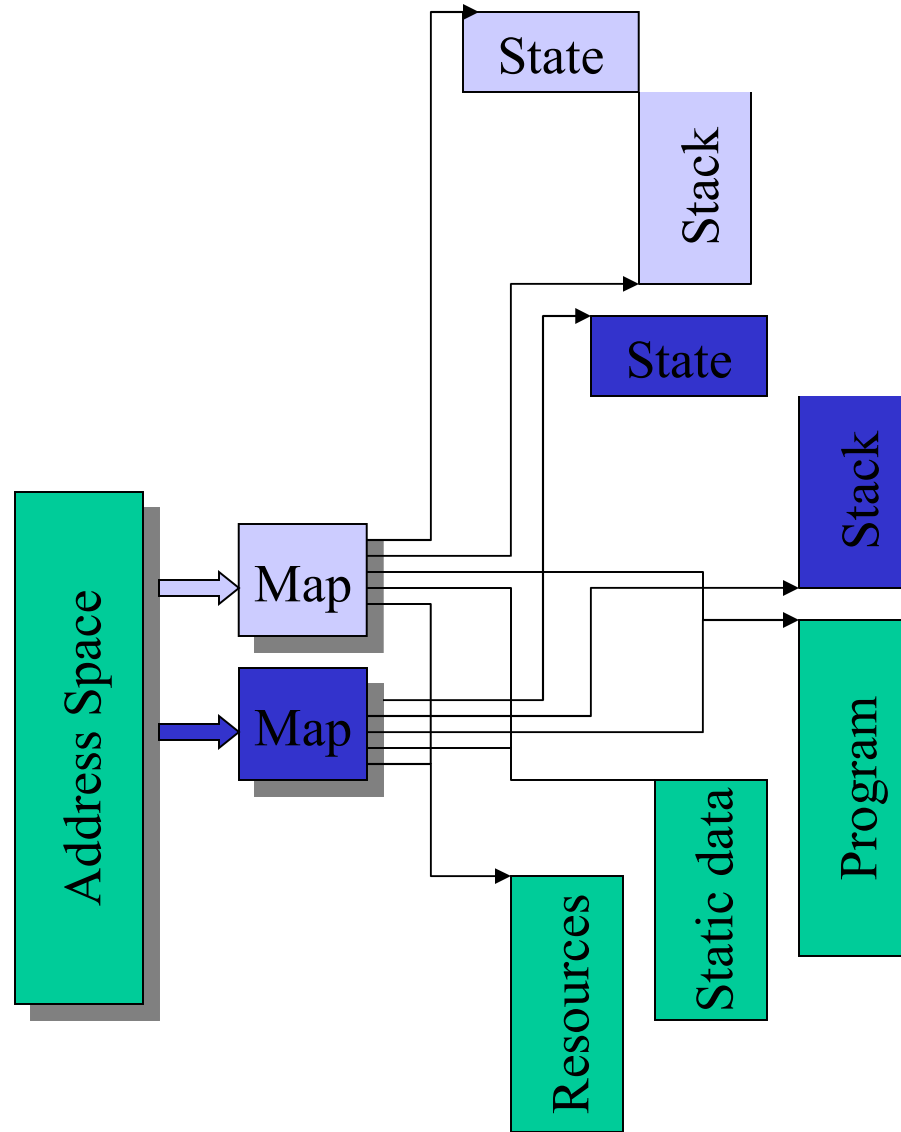


- Define & implement the essential characteristics of a process and thread
 - Algorithms to define the behavior
 - Data structures to preserve the state of the execution
- Define what “things” threads in the process can reference – the *address space* (most of the “things” are memory locations)
- Manage the resources used by the processes/threads
- Tools to create/destroy/manipulate processes & threads
- Tools to time-multiplex the CPU – Scheduling the (Chapter 7)
- Tools to allow threads to synchronization the operation with one another (Chapters 8-9)
- Mechanisms to handle deadlock (Chapter 10)
- Mechanisms to handle protection (Chapter 14)

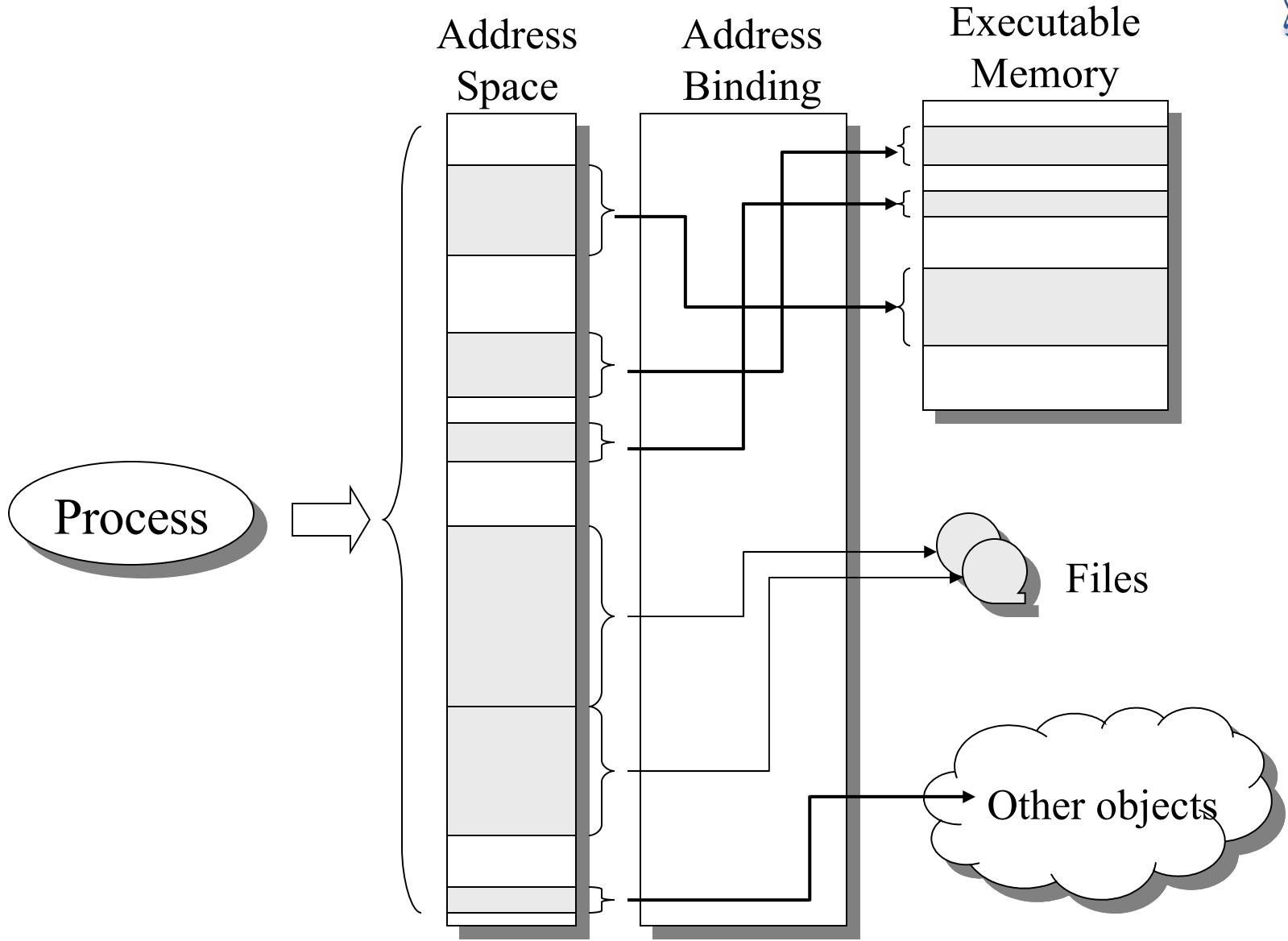
Modern Processes and Threads



Processes & Threads



The Address Space



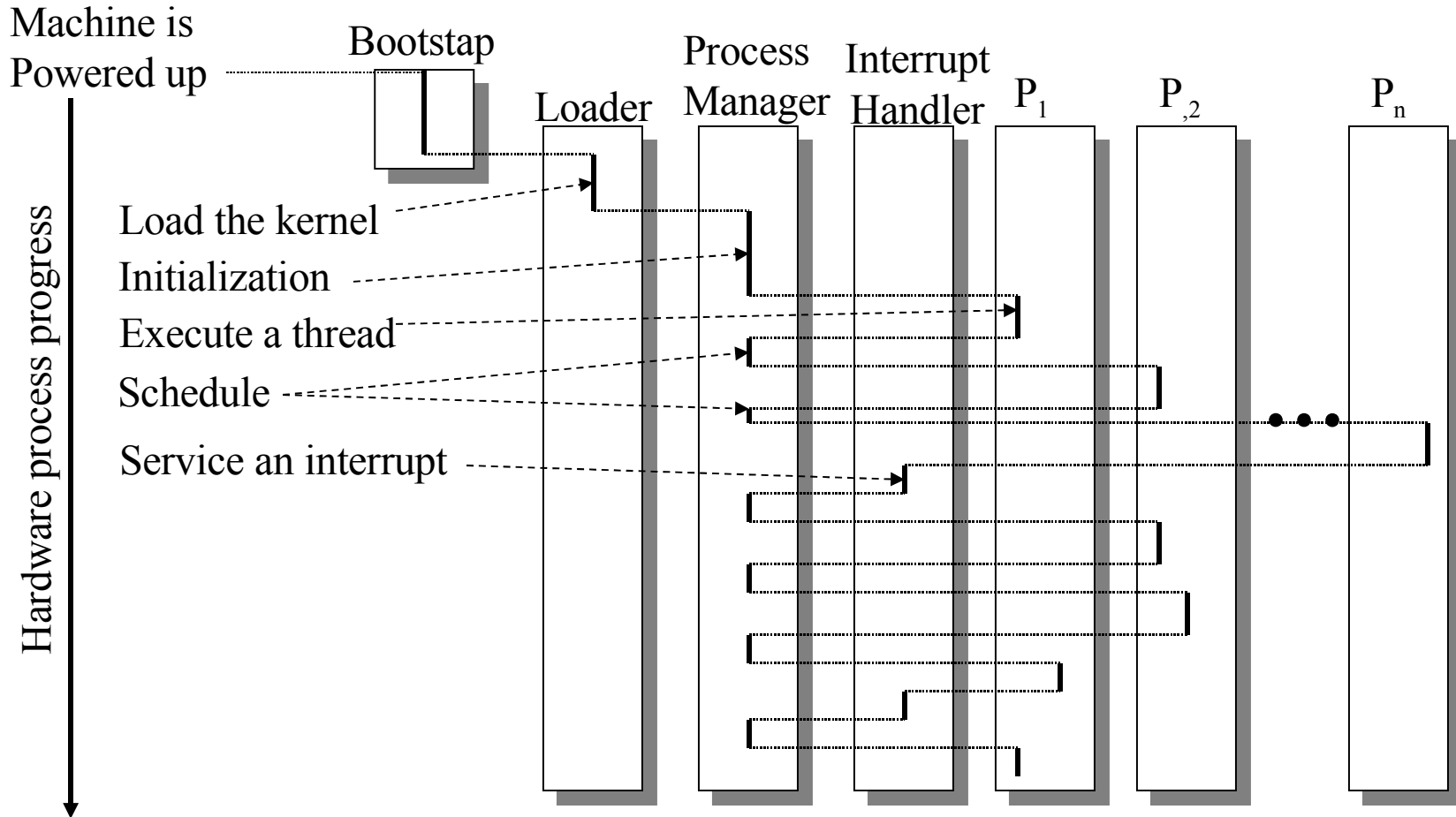
Building the Address Space



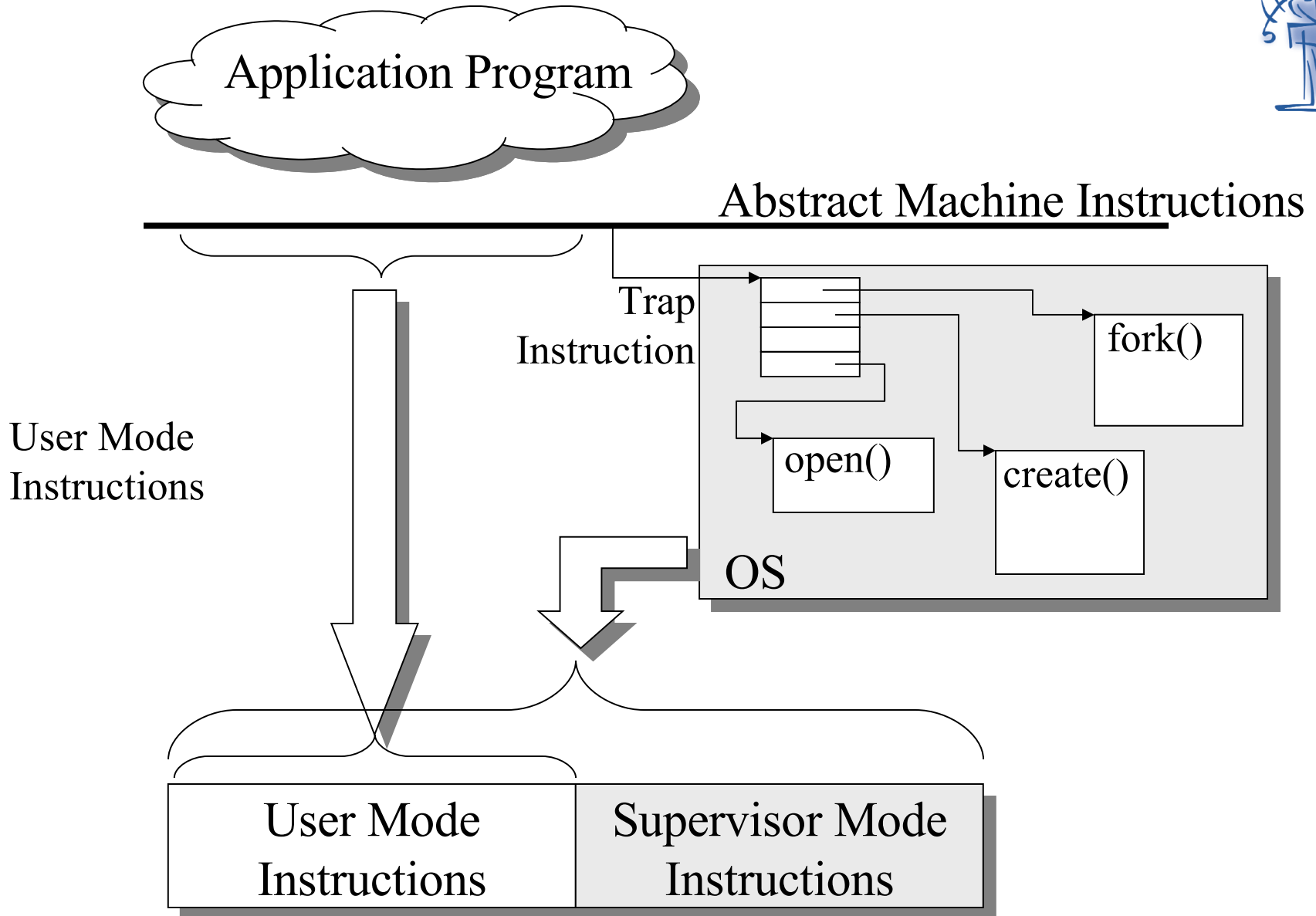
- Some parts are built into the environment
 - Files
 - System services
- Some parts are imported at runtime
 - Mailboxes
 - Network connections
- Memory addresses are created at compile (and run) time



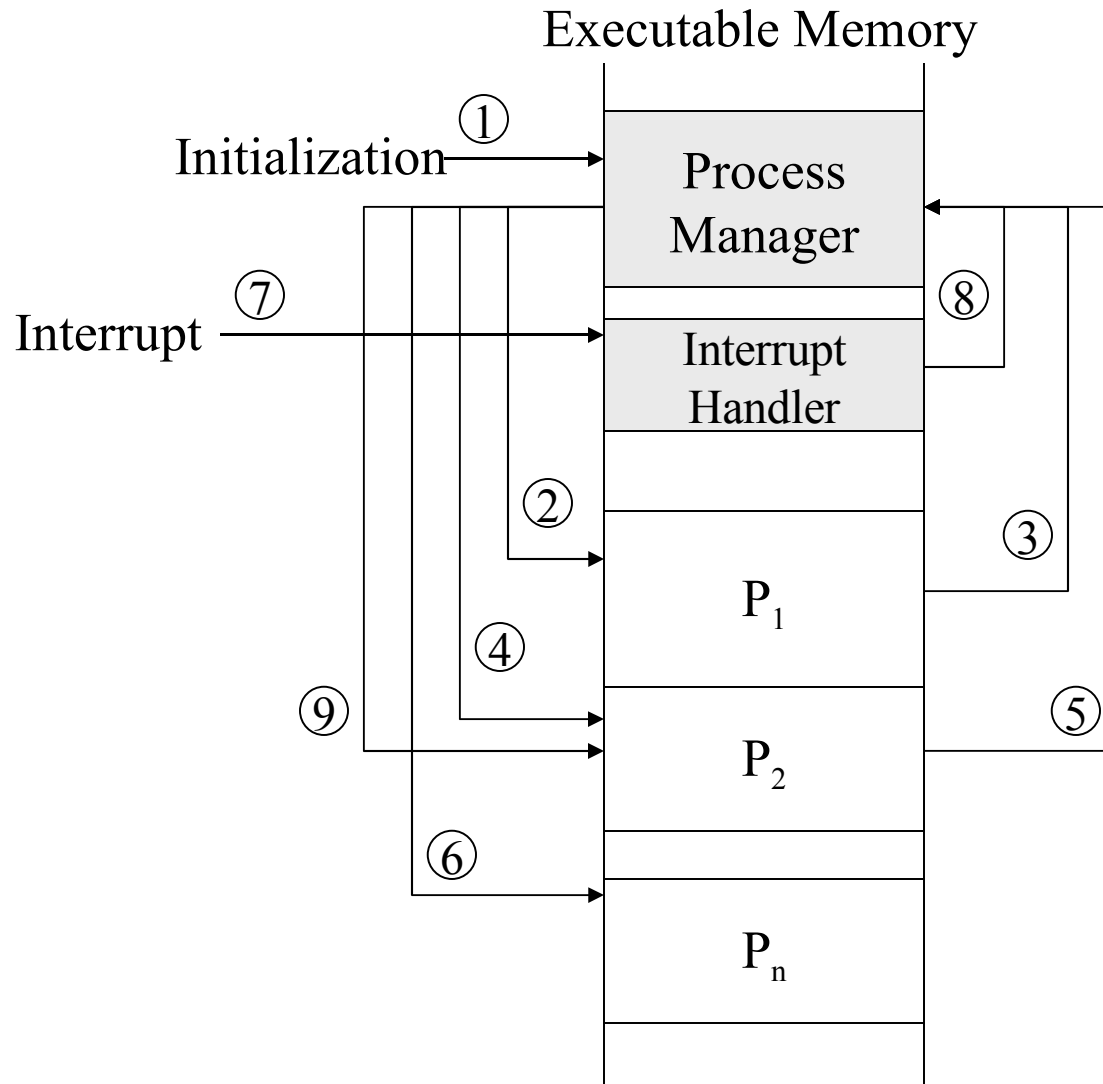
Tracing the Hardware Process



The Abstract Machine Interface



Context Switching

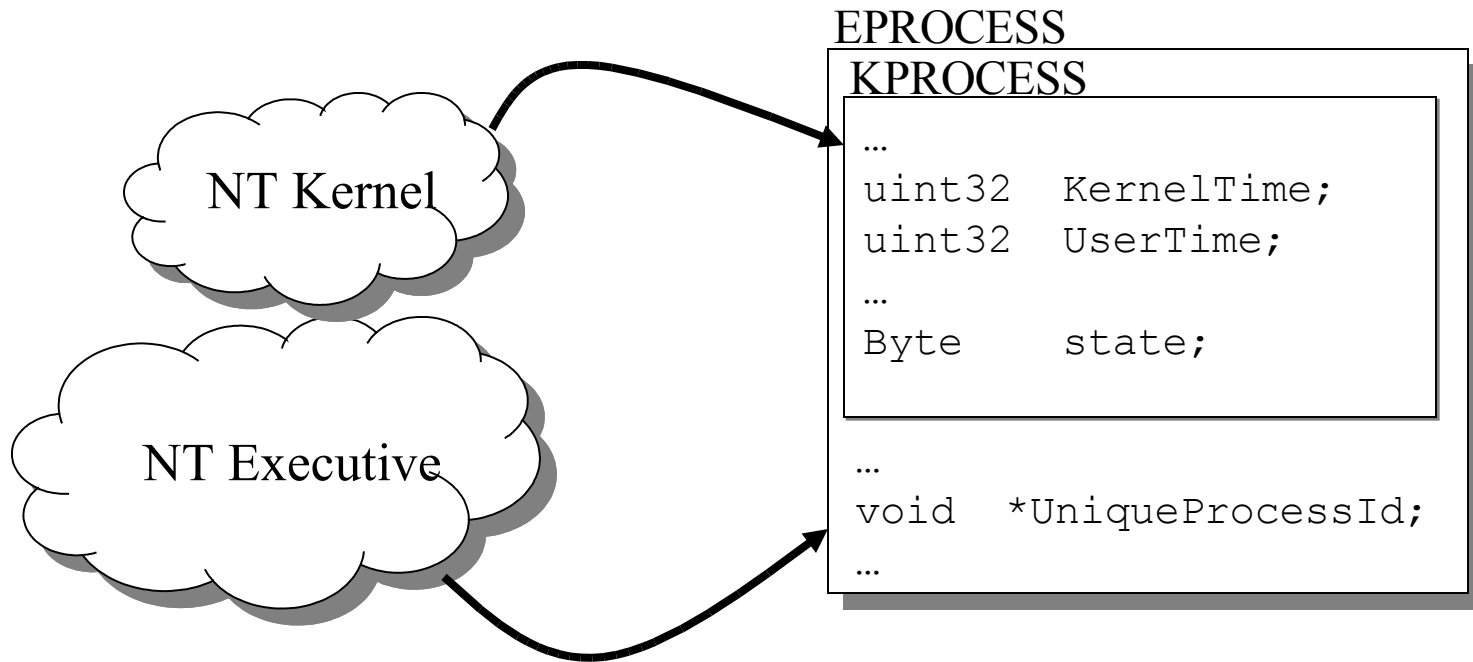




Process Descriptors

- OS creates/manages process abstraction
- Descriptor is data structure for each process
 - Register values
 - Logical state
 - Type & location of resources it holds
 - List of resources it needs
 - Security keys
 - etc. (see Table 6.1 and the source code of your favorite OS)

Windows NT Process Descriptor





Windows NT Process Descriptor (2)

Kernel process object including:

- Pointer to the page directory

- Kernel & user time

- Process base priority

- Process state

- List of the Kernel thread descriptors that are using this process



Windows NT Process Descriptor (3)

Parent identification

Exit status

Creation and termination times.

Memory status

Security information

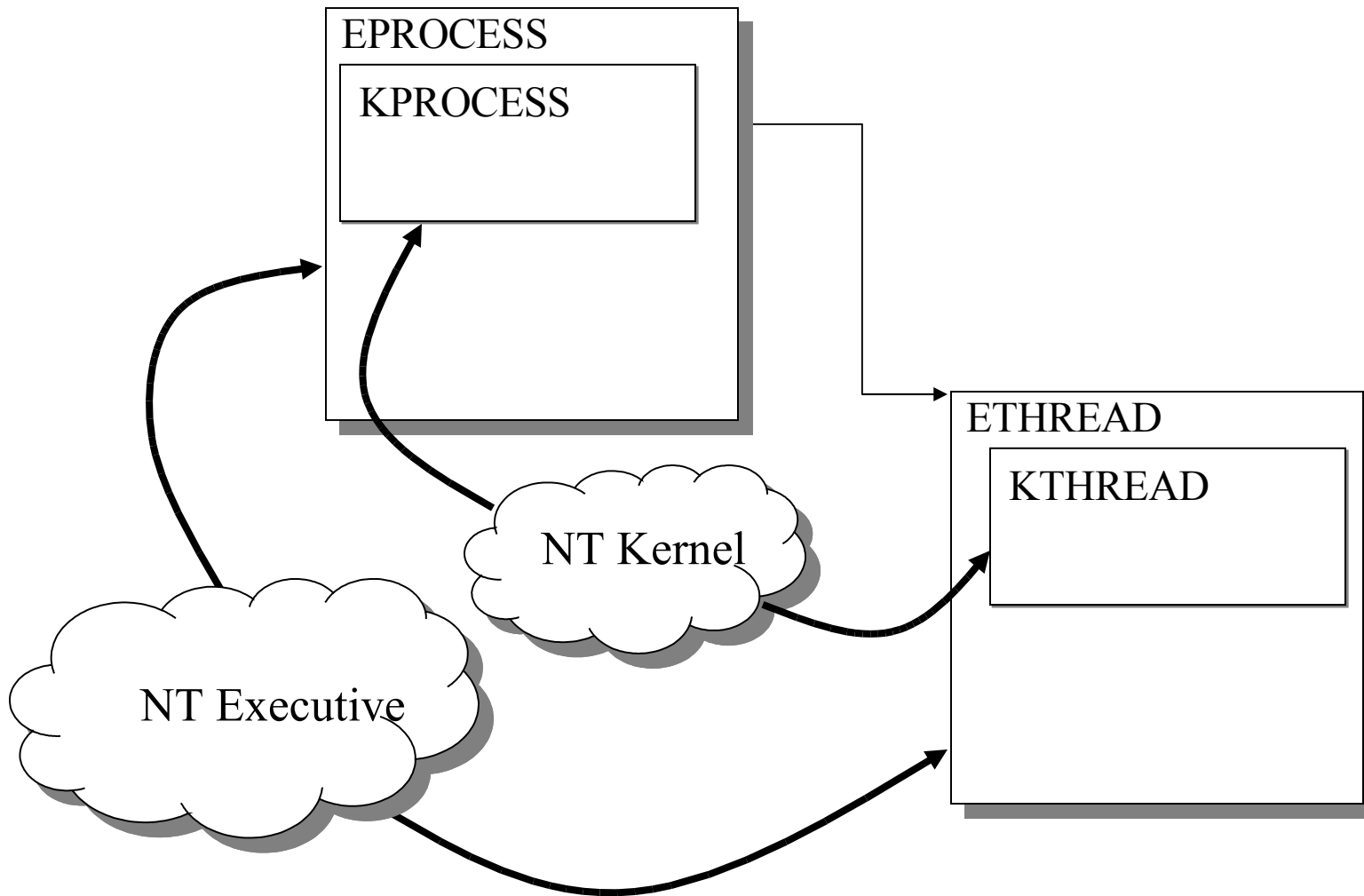
executable image

Process priority class used by the thread scheduler.

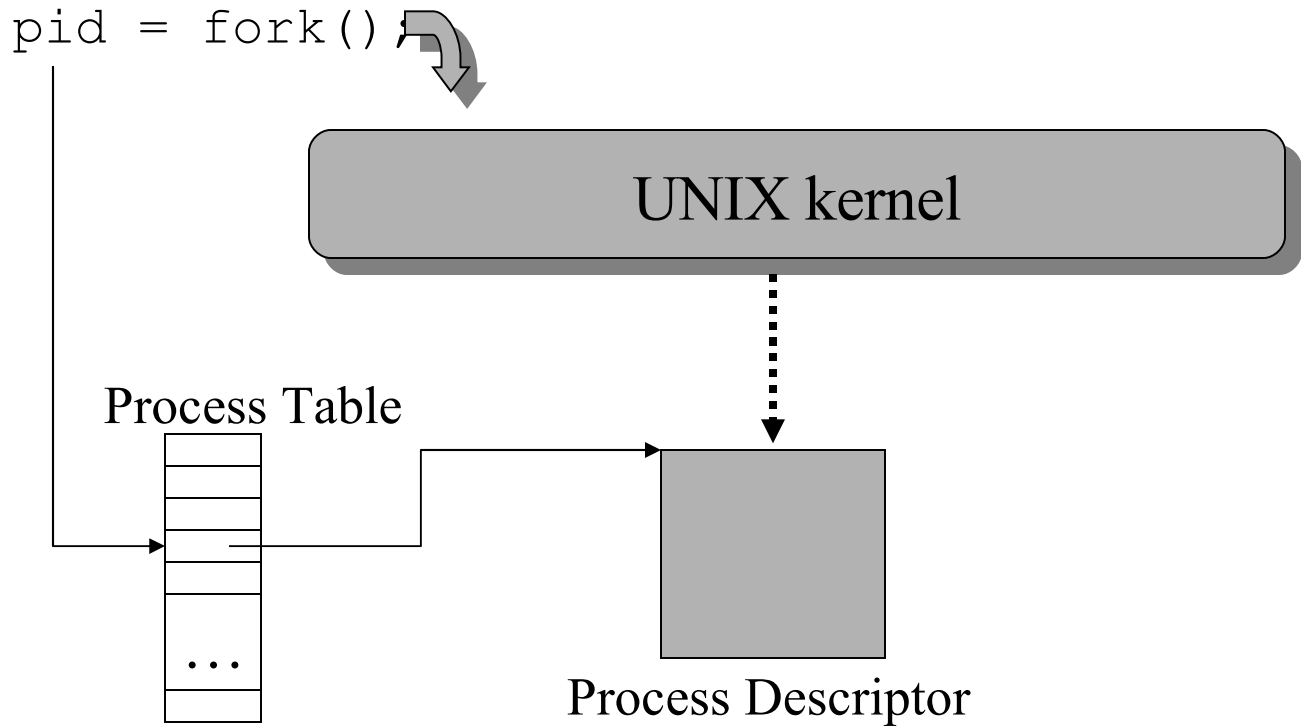
A list of handles used by this process

A pointer to Win32-specific information

Windows NT Thread Descriptor



Creating a Process in UNIX



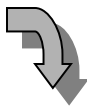
Creating a Process in NT

CreateProcess (...);



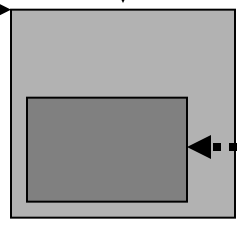
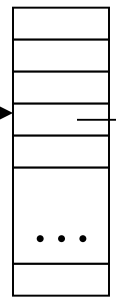
Win32 Subsystem

ntCreateProcess (...);
...
ntCreateThread (...);



NT Executive

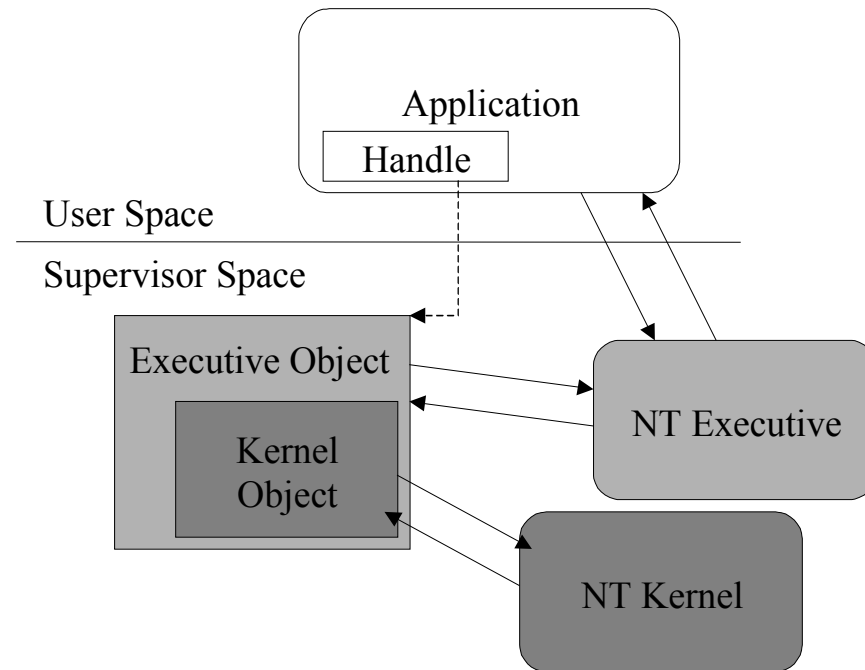
Handle Table



NT Kernel

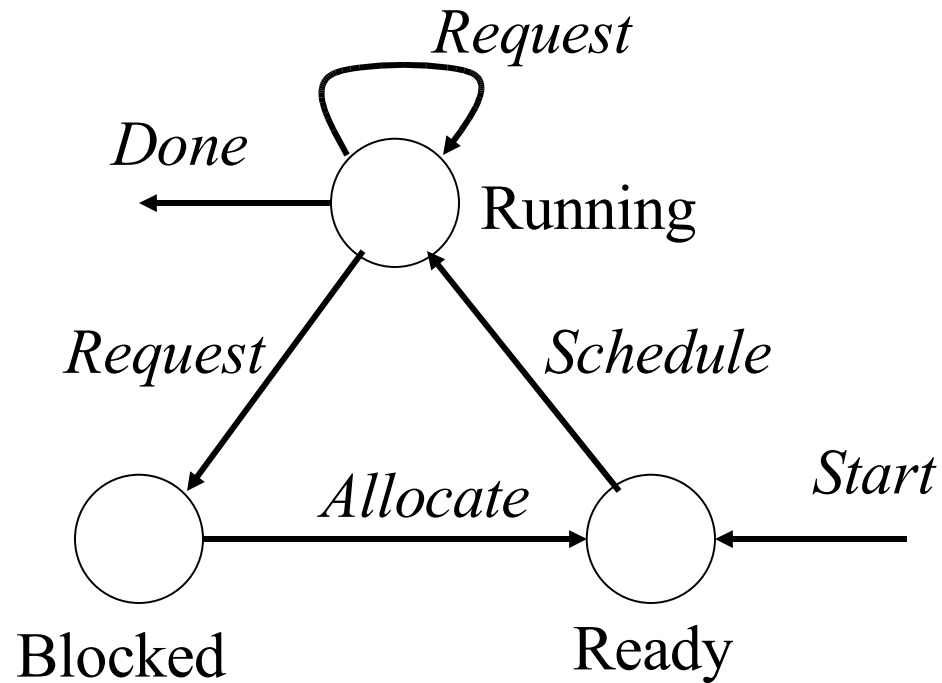
Process Descriptor

Windows NT Handles

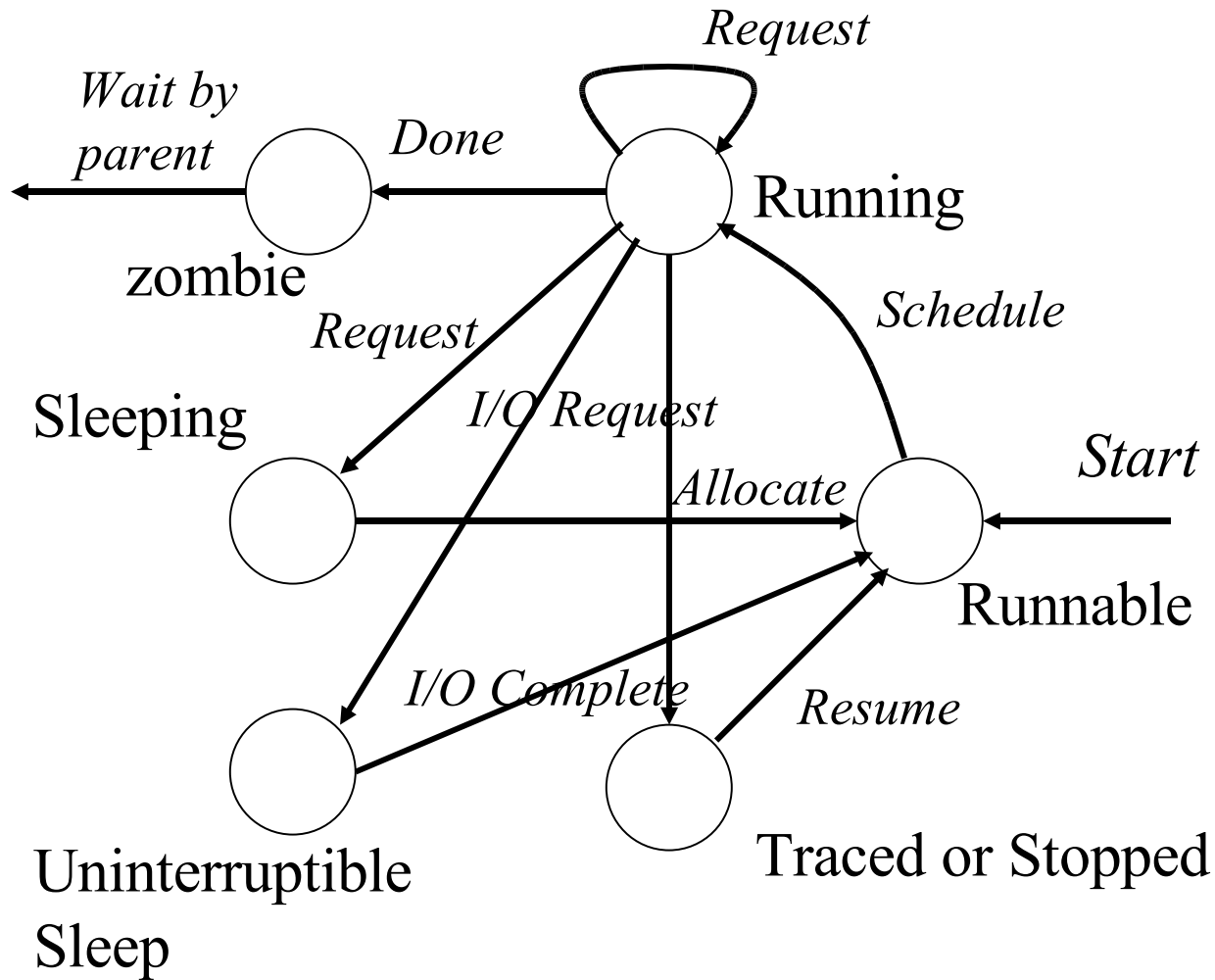




Simple State Diagram

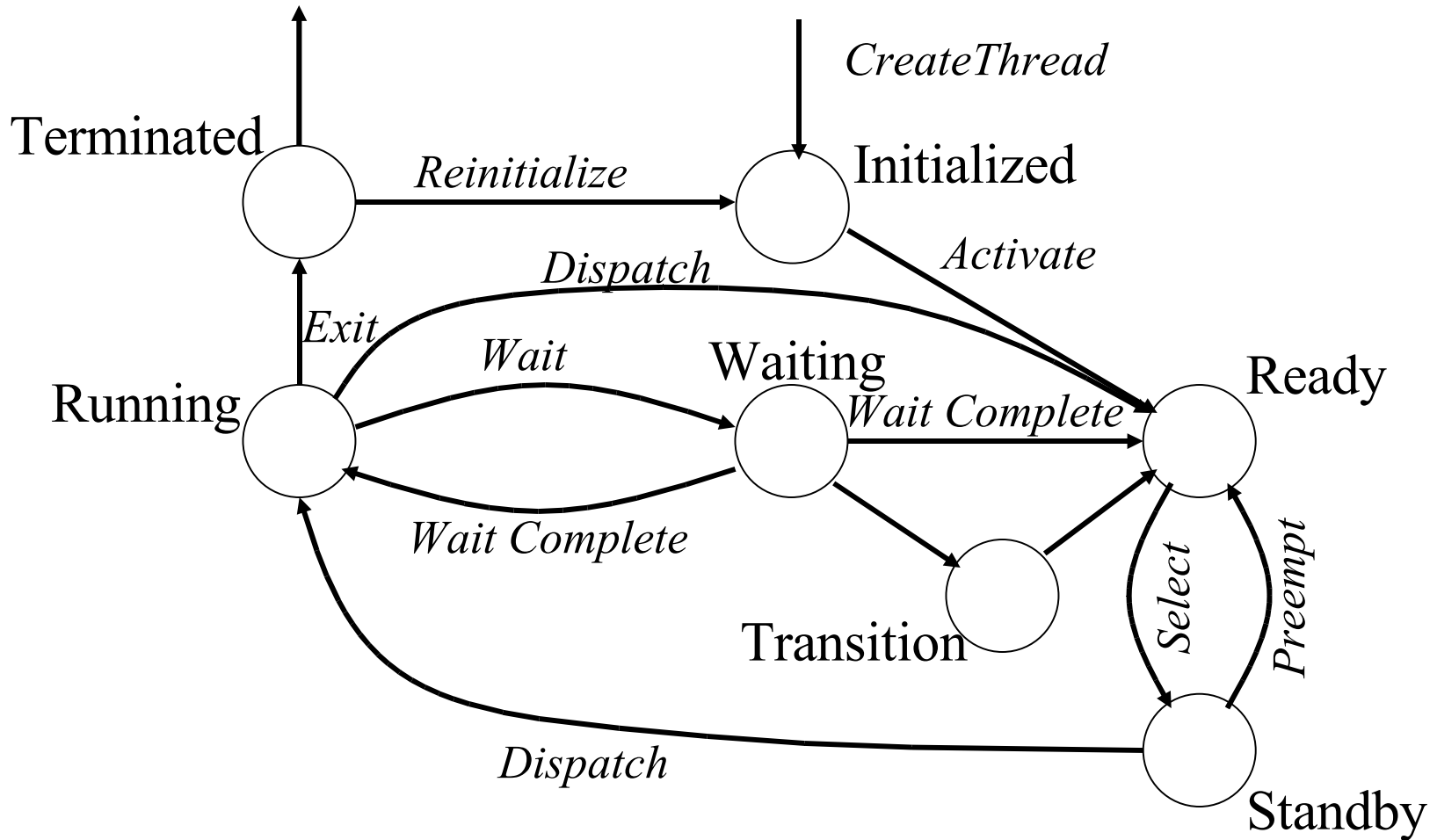


UNIX State Transition Diagram





Windows NT Thread States





Resources

Resource: Anything that a process can request, then be blocked because that thing is not available.

$R = \{R_j \mid 0 \leq j < m\}$ = resource types

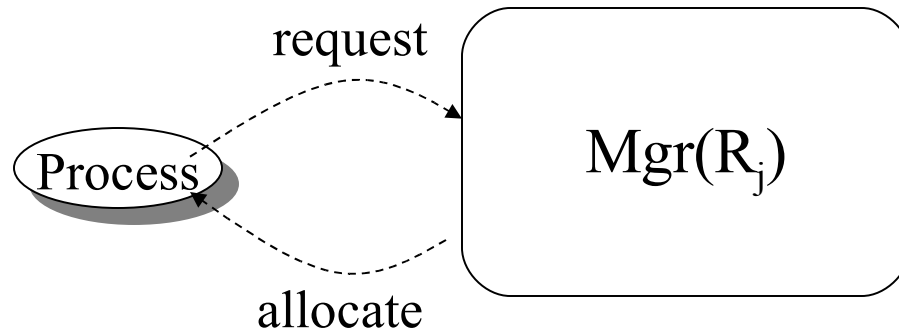
$C = \{c_j \mid 0 \leq c_j \leq R_j \ (0 \leq j < m)\}$ = units of R_j available

Reusable resource: After a unit of the resource has been allocated, it must ultimately be released back to the system. E.g., CPU, primary memory, disk space, ... The maximum value for c_j is the number of units of that resource

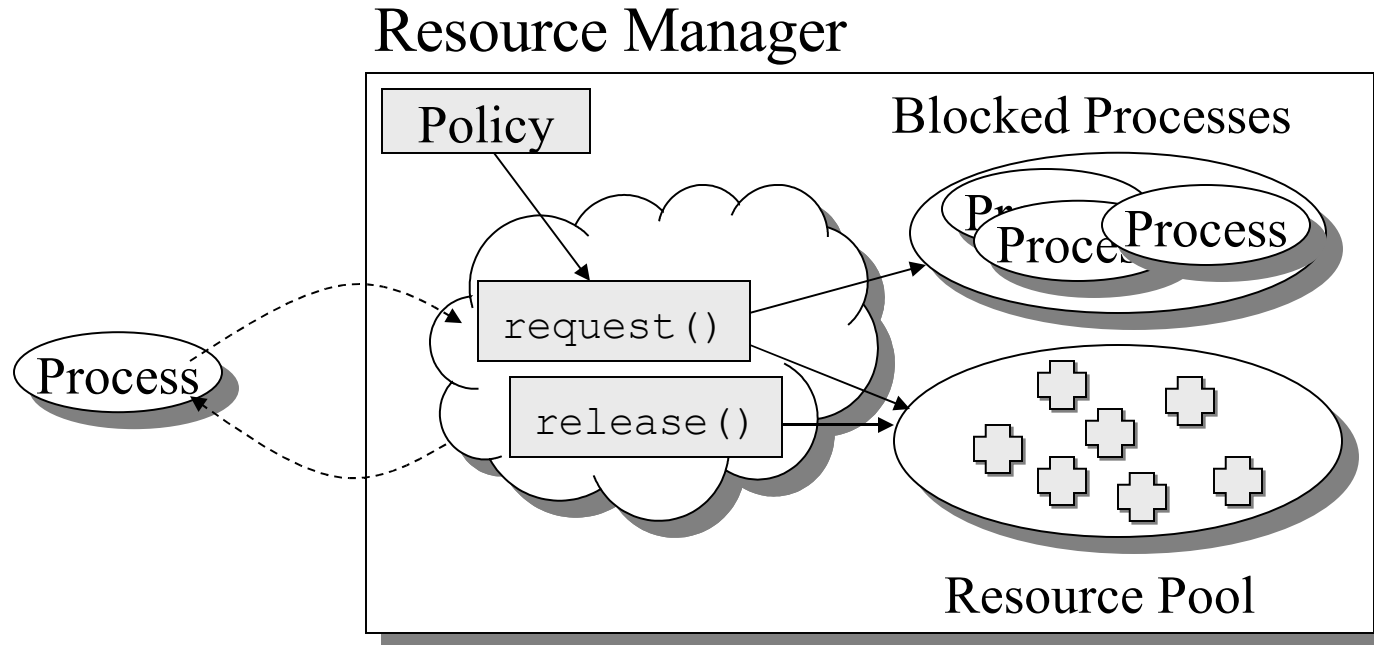
Consumable resource: There is no need to release a resource after it has been acquired. E.g., a message, input data, ... Notice that c_j is unbounded.

Using the Model

- There is a resource manager, $\text{Mgr}(R_j)$ for every R_j
- Process p_i can request units of R_j if it is currently *running*
 - p_i can only request $n_i - c_j$ units of reusable R_j
 - p_i can request unbounded # of units of consumable R_j
- $\text{Mgr}(R_j)$ can allocate units of R_j to p_i

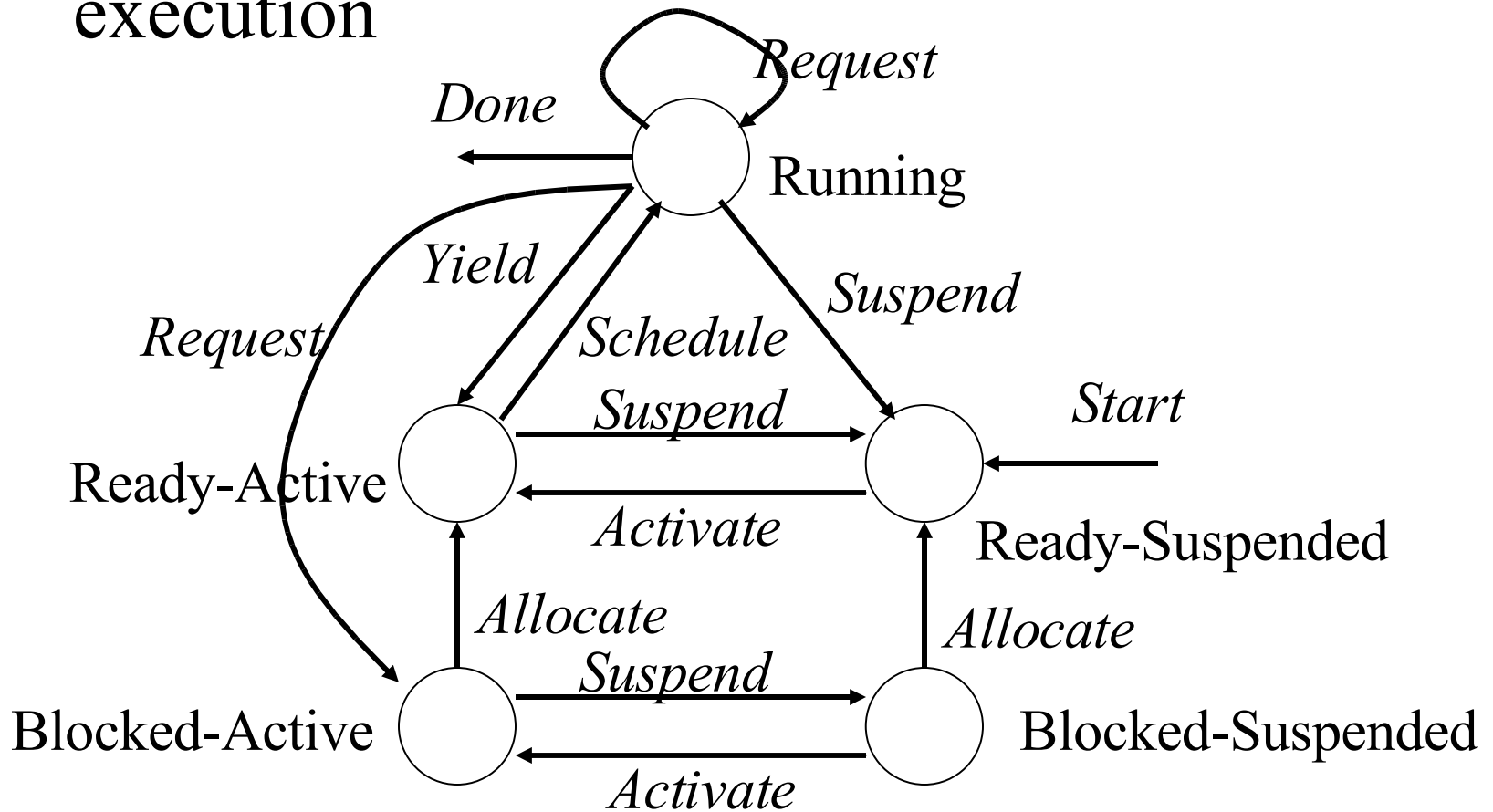


A Generic Resource Manager



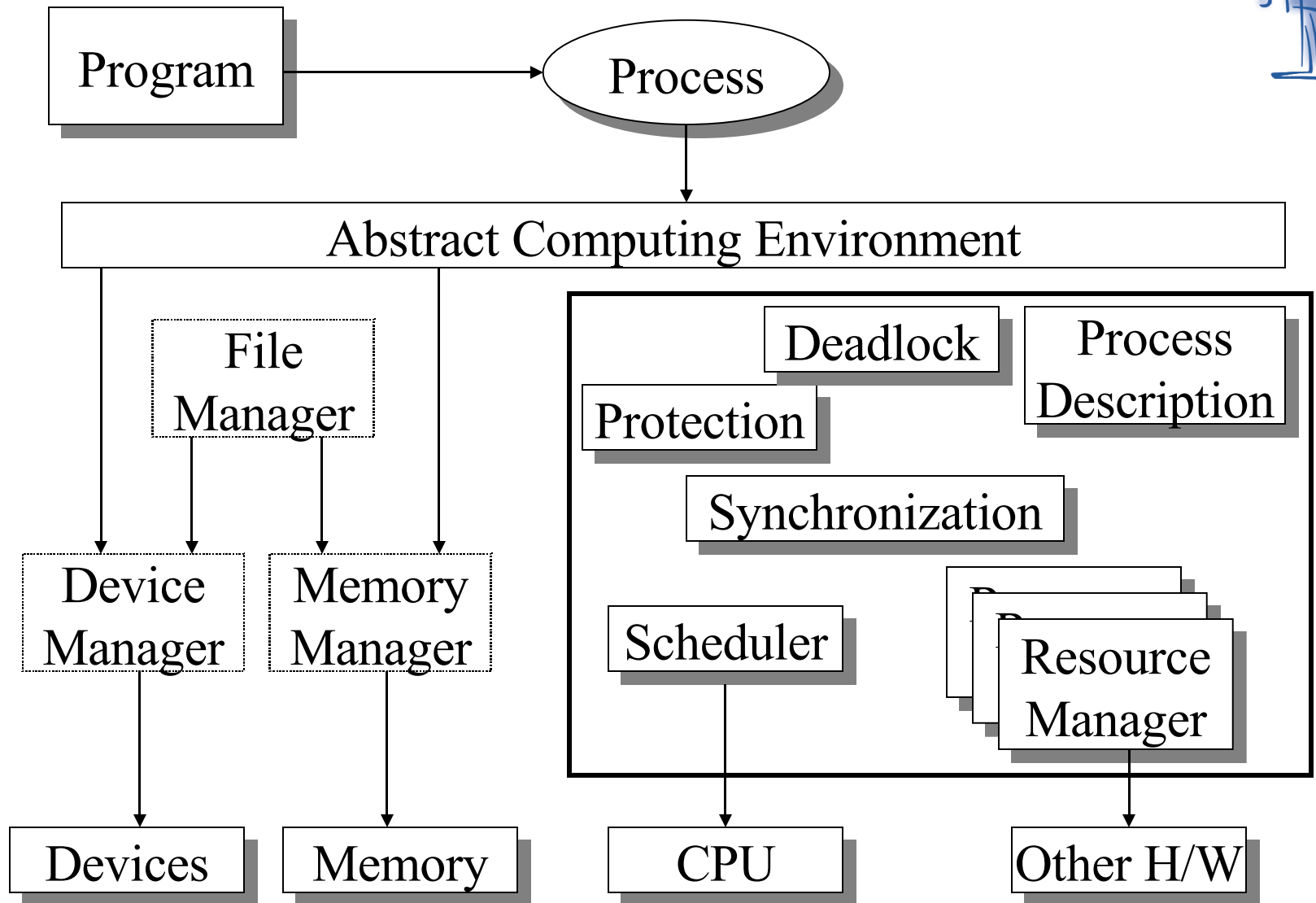
Process Hierarchies

- Parent-child relationship may be significant: parent controls children's execution



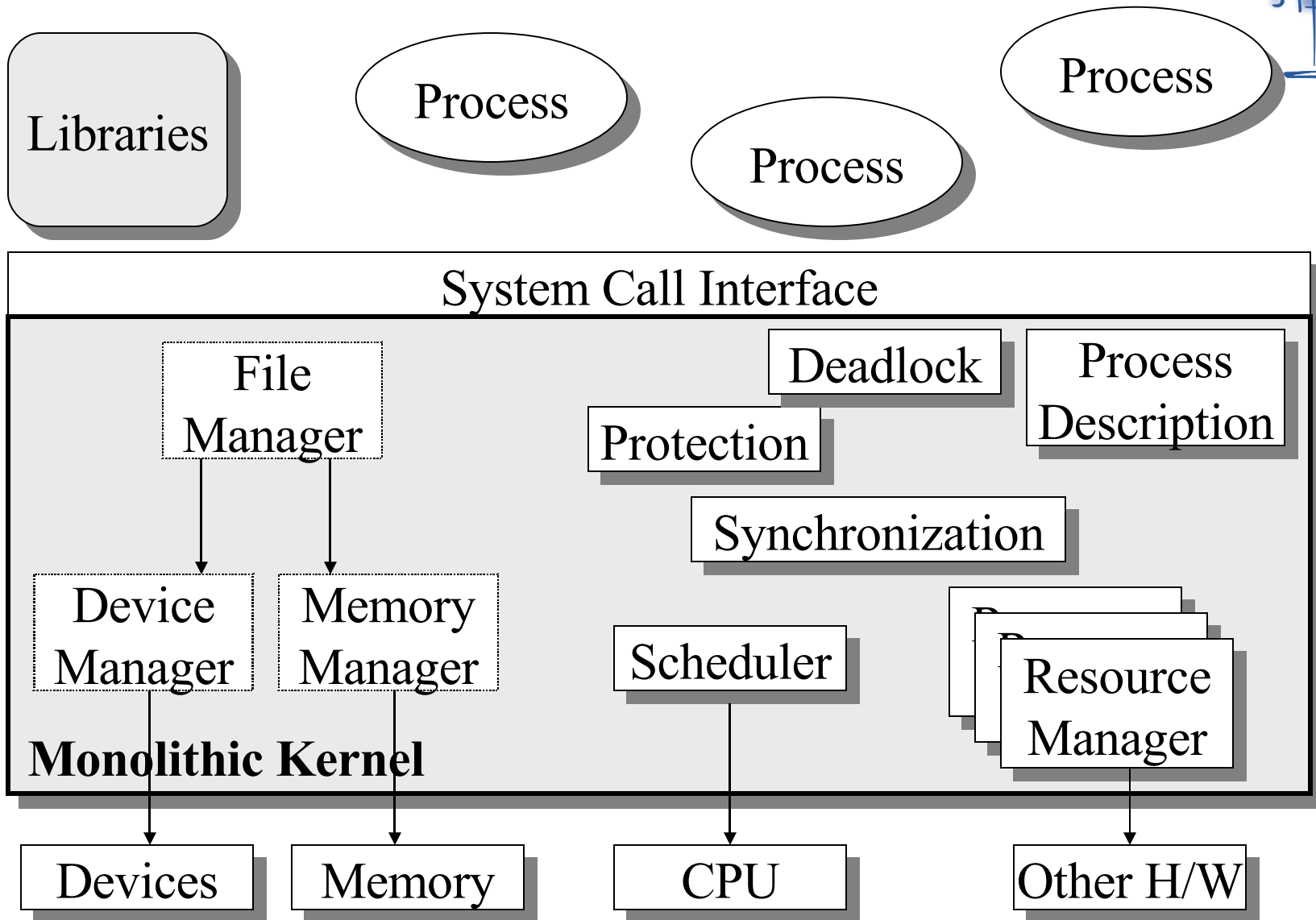


Process Manager Overview





UNIX Organization





Windows NT Organization

