

OS Organization

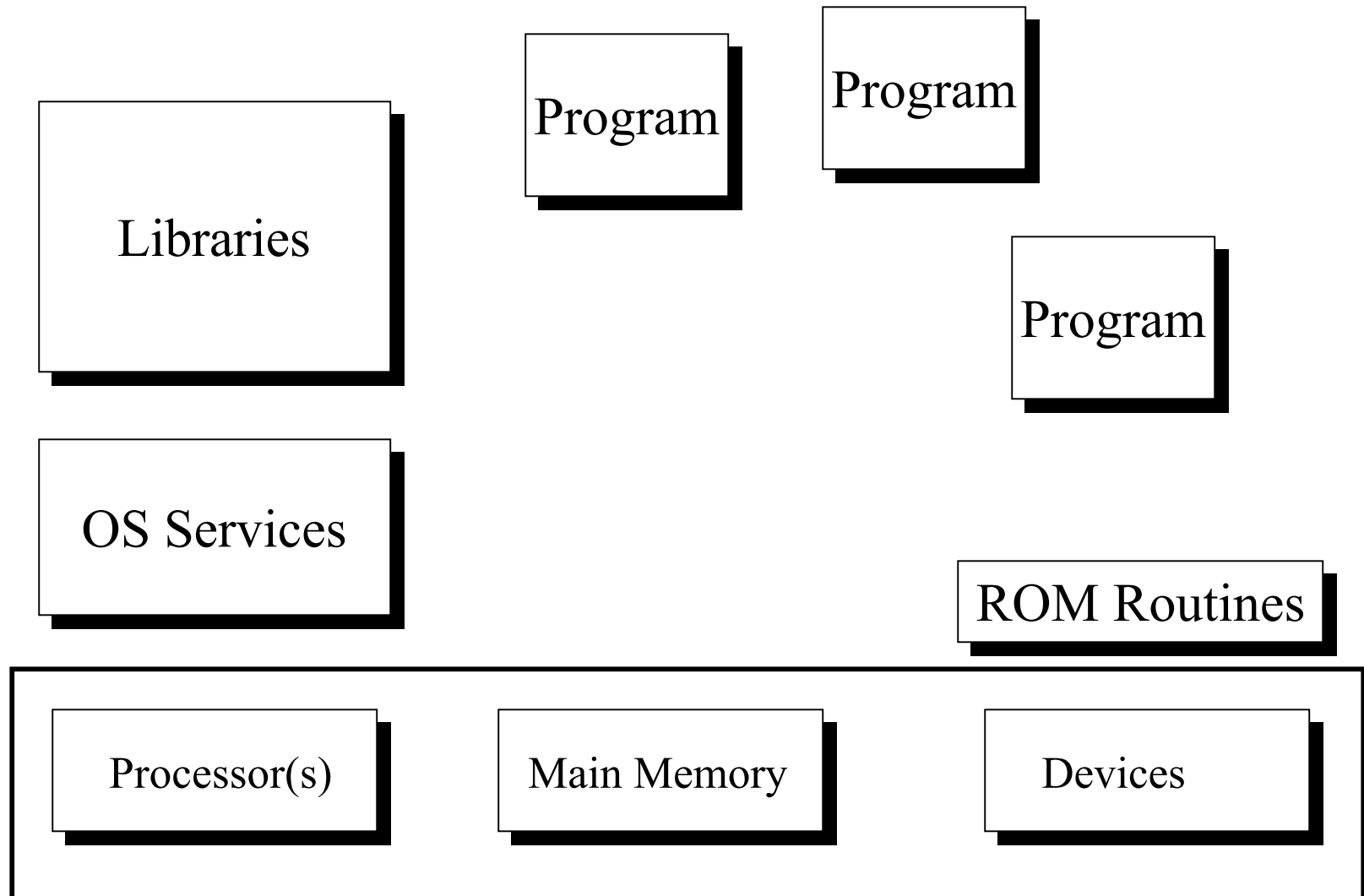
OS Requirements

- Provide resource abstractions

Performance

- The OS is an overhead function => should not use too much of machine's resources
- Minimum functionality is to implement abstractions
- Additional function must be traded off against performance
 - DOS: one process
 - UNIX: low level file system

Single Process OS Organization



OS Requirements

- Provide resource abstractions
- Provide process abstraction
- Manage sharing
- Ensure isolation between processes

Constraints

- Resource abstraction
- Performance
- Protection and security
- Correctness
- Maintainability
- Commercial factors
- Standards and open systems

Protection & Security

- Multiprogramming => resource sharing
- Therefore, need software-controlled resource isolation
- Security policy: Sharing strategy chosen by computer's owner
- Protection mechanism: Tool to implement a family of security policies

Correctness & Maintainability

- Security depends on correct operation of software => trusted vs untrusted software
- Maintainability relates to ability of software to be changed
- If either is sufficiently important, can limit the function of the OS
 - Guiding a manned spaceship
 - Managing a nuclear reactor

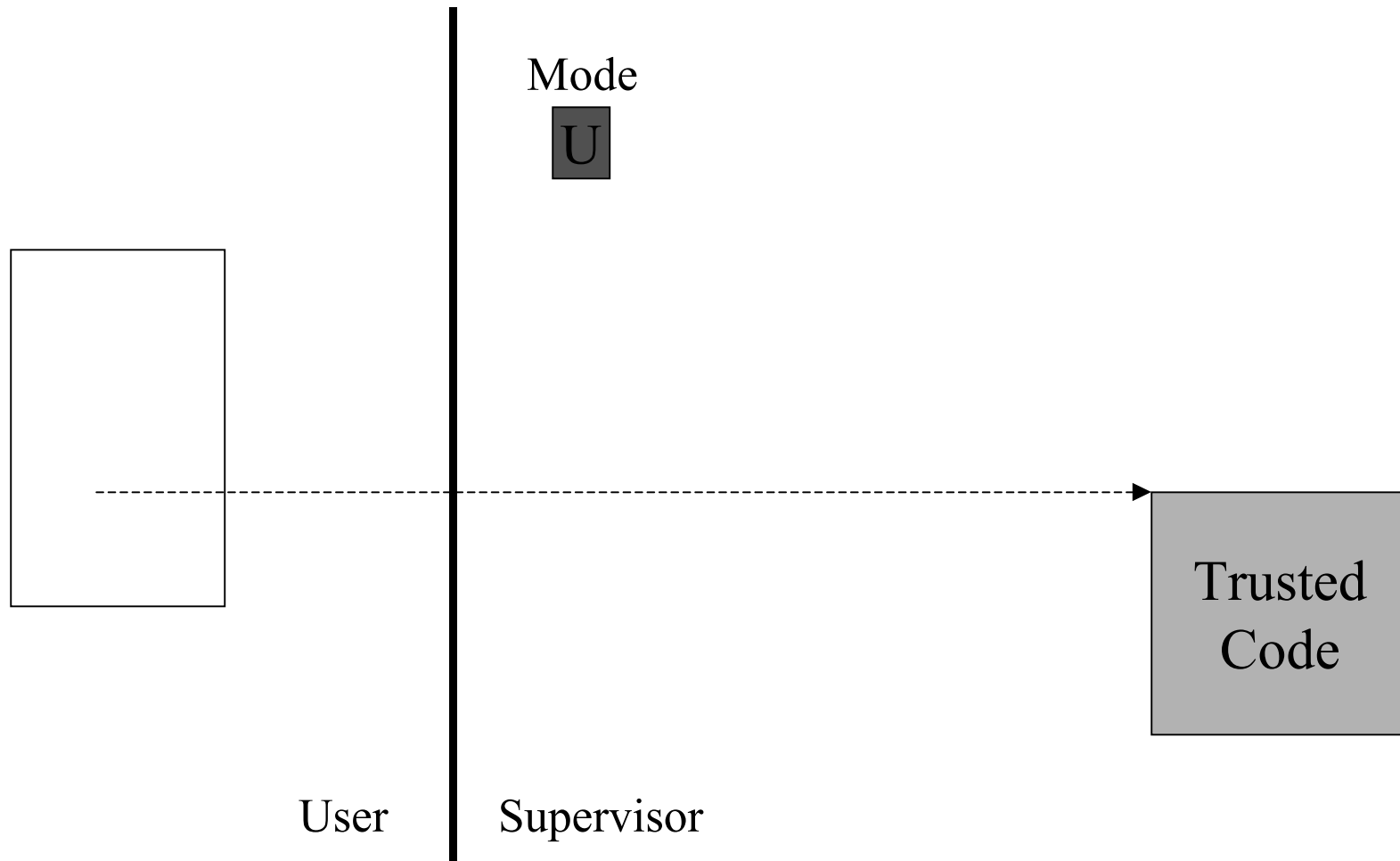
Processor Modes

- Mode bit: Supervisor or User mode
- Supervisor mode
 - Can execute all machine instructions
 - Can reference all memory locations
- User mode
 - Can only execute a subset of instructions
 - Can only reference a subset of memory locations

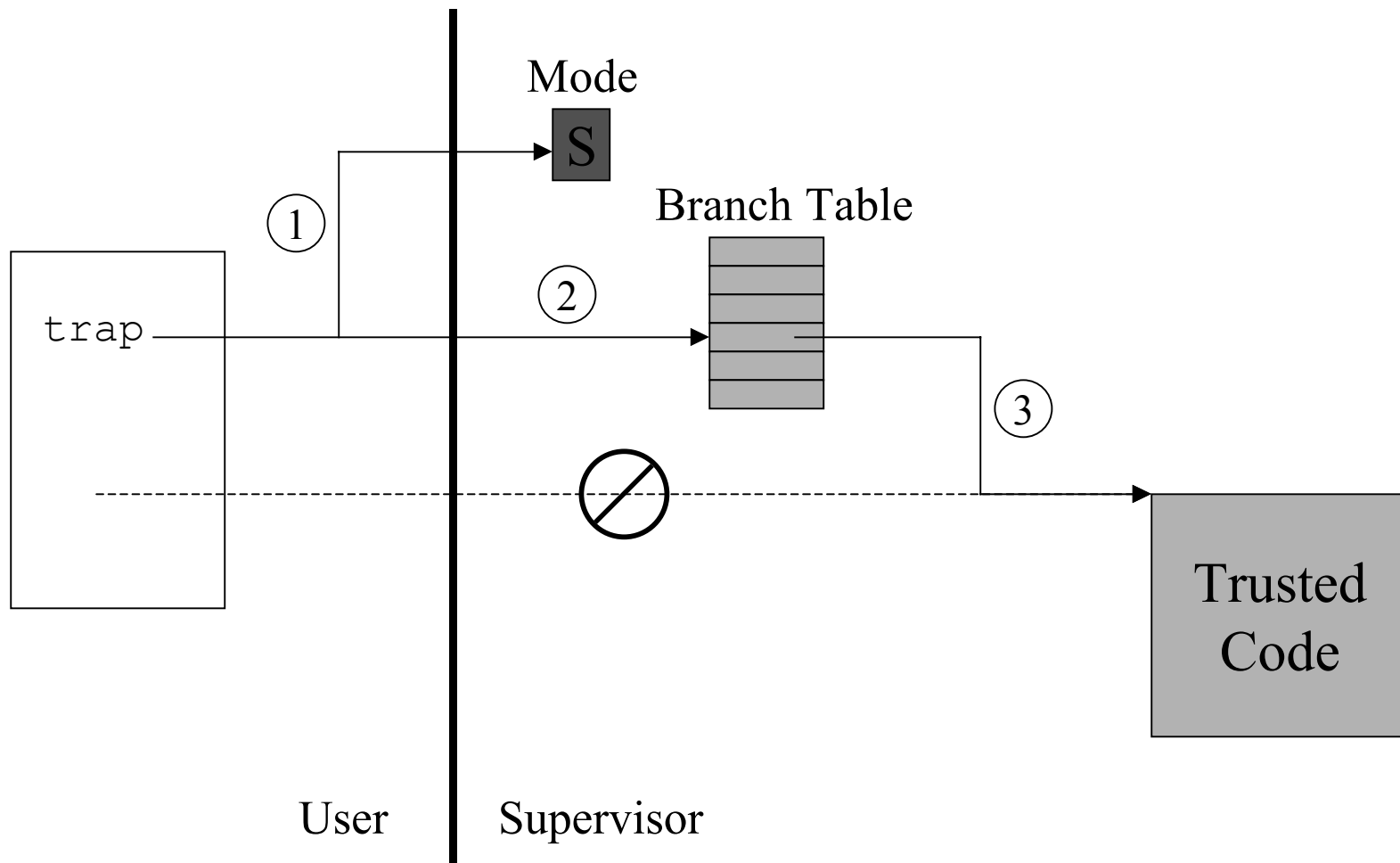
Kernels

- The part of the OS critical to correct operation (trusted software)
- Executes in supervisor mode
- The `trap` instruction is used to switch from user to supervisor mode, entering the OS

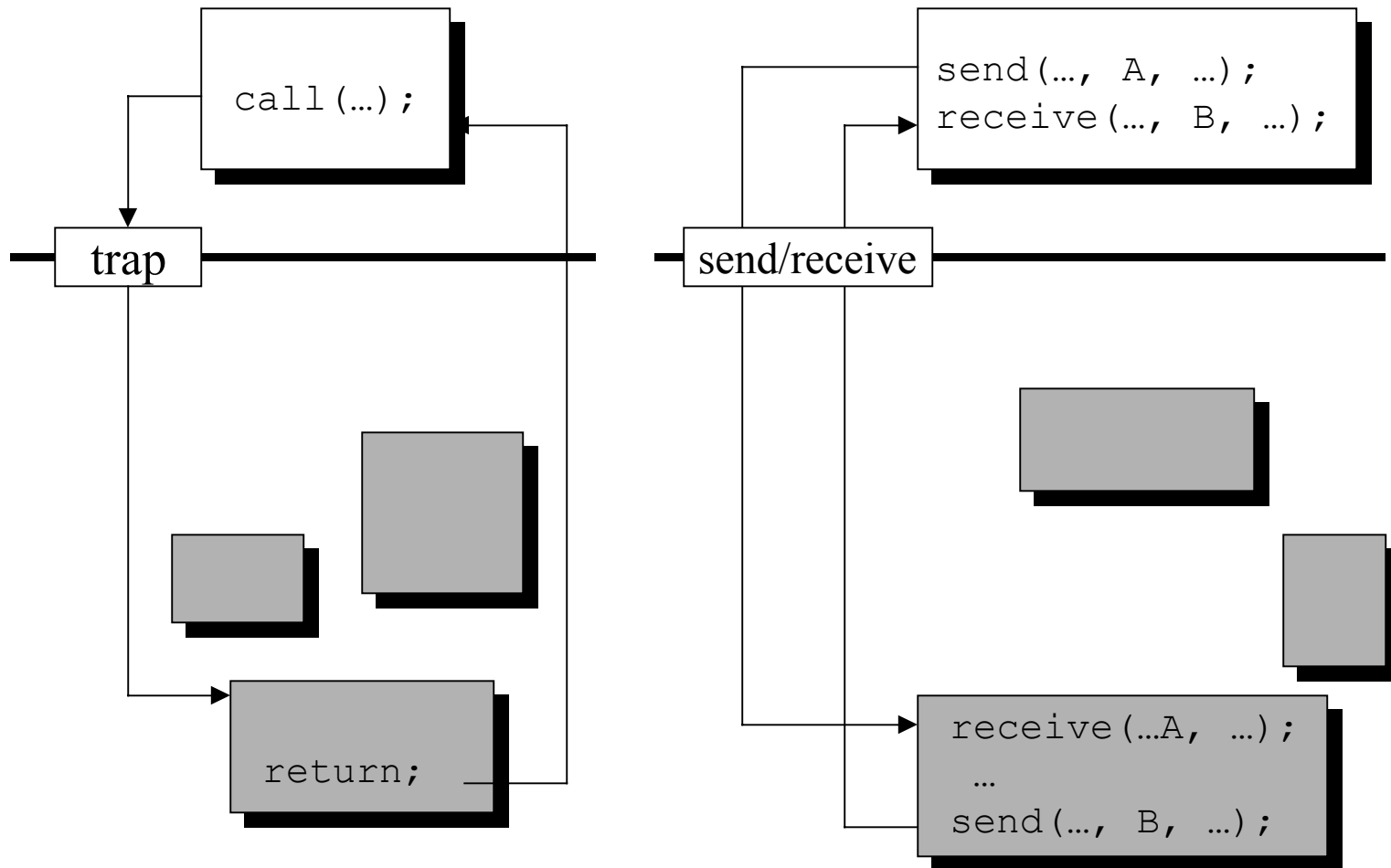
The trap Instruction



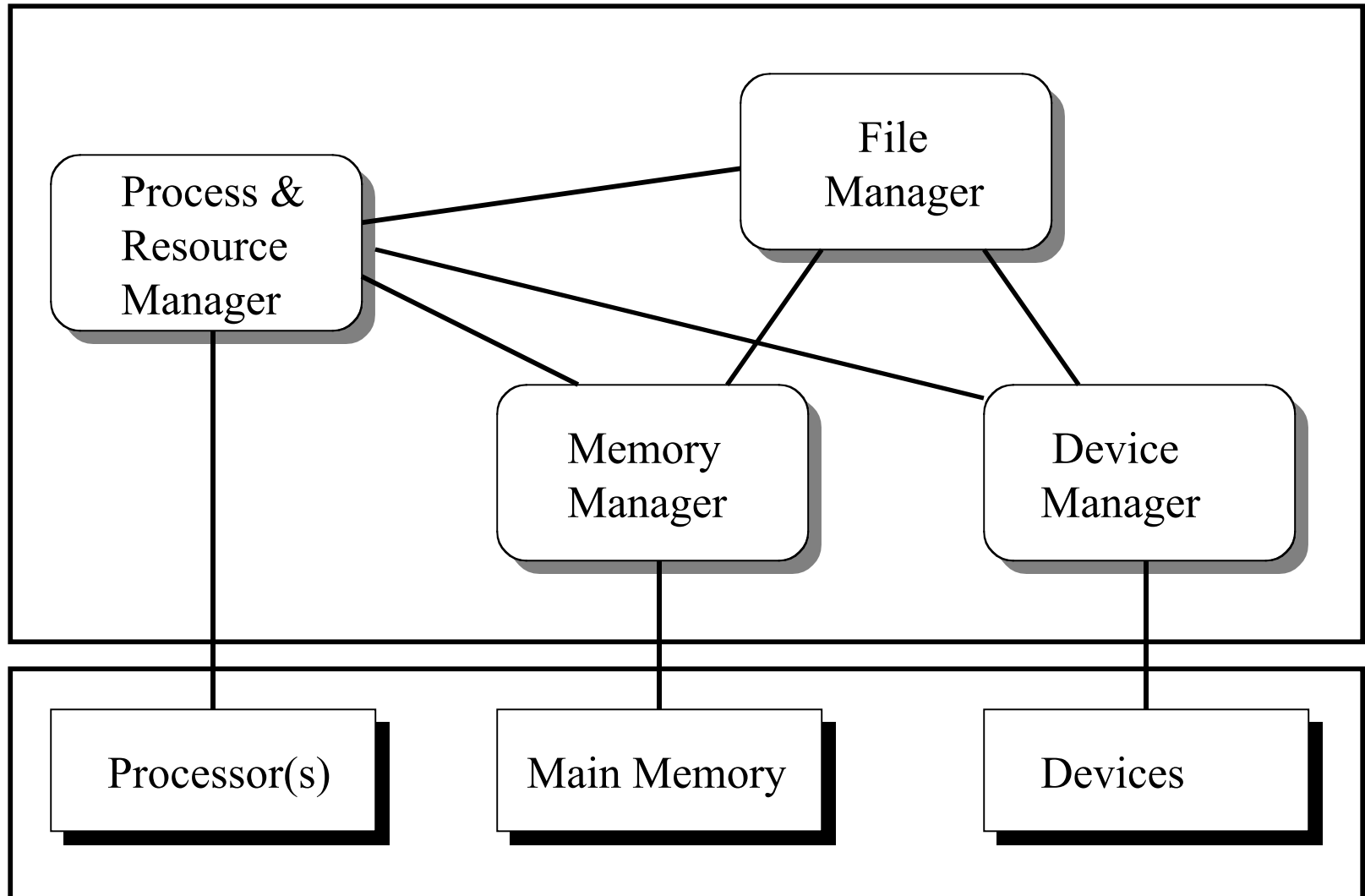
The trap Instruction



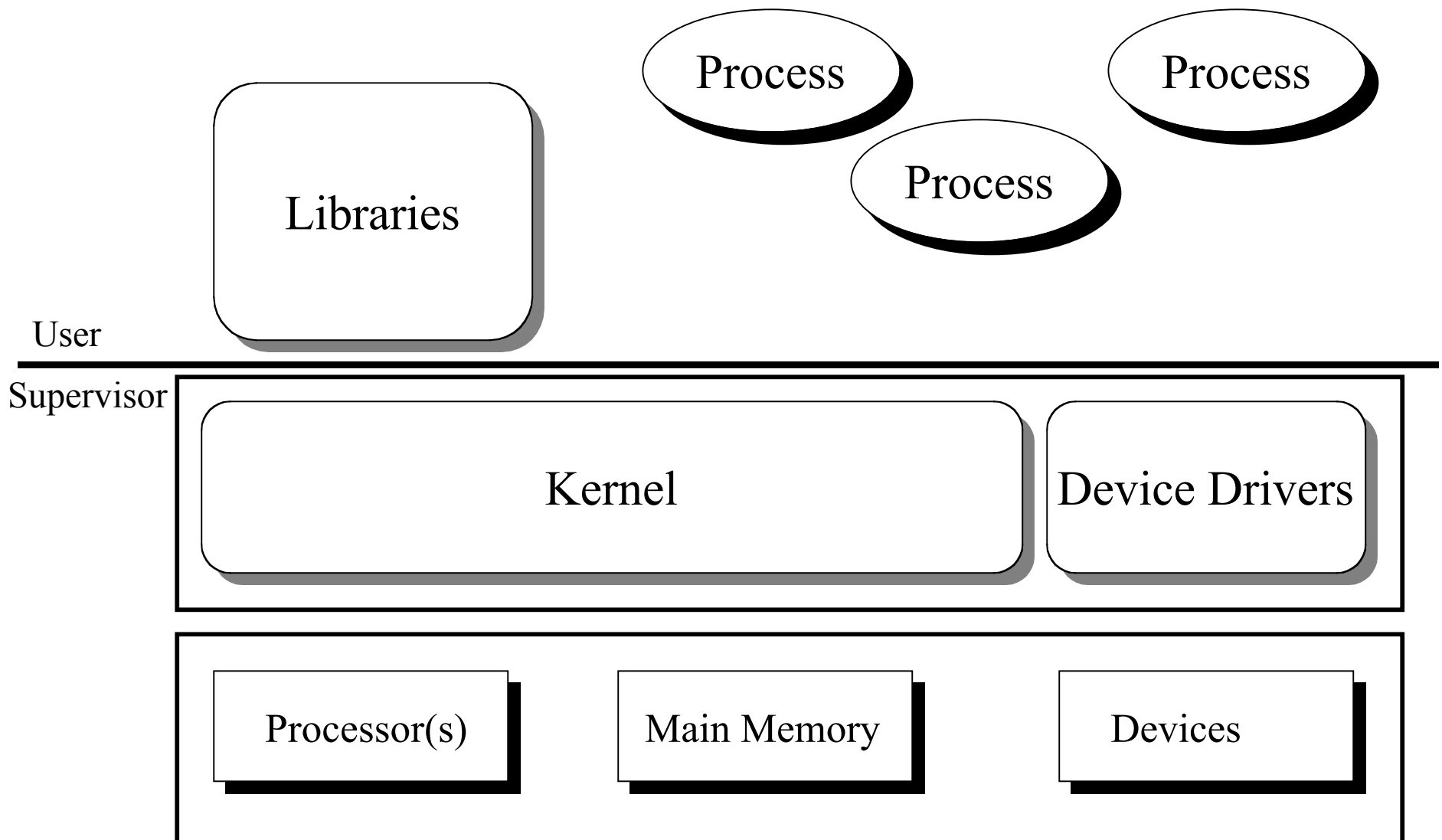
Requesting OS Service



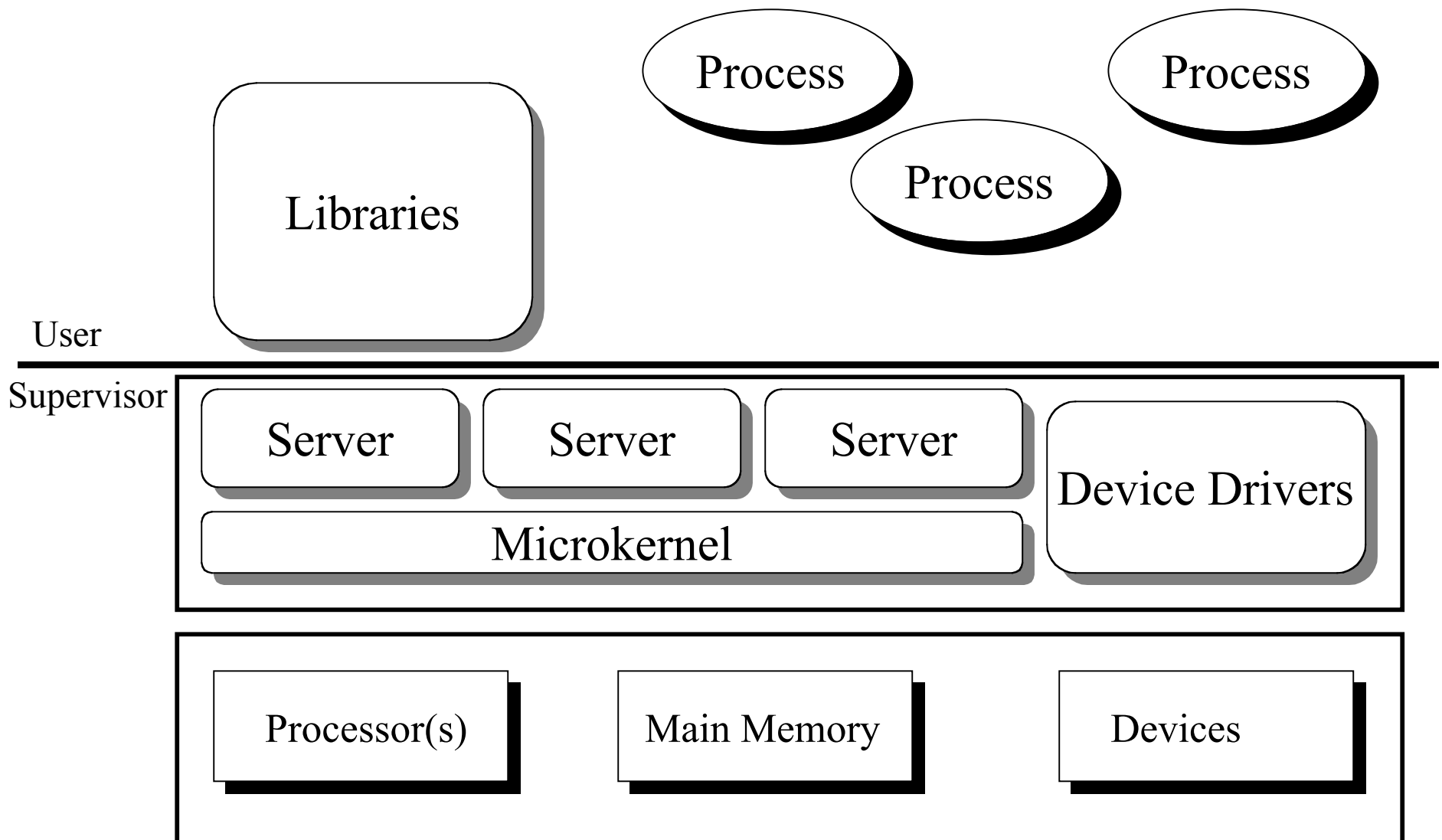
Basic OS Organization



The UNIX Organization



Microkernel Organization



NT Organization

