Operating Systems - Chapter 15 Summary

Mechanism: Address Translation

• Extends limited direct execution to memory virtualization.
• Goals: Efficiency (fast translation), Control (protection), Flexibility (ease of programming).
• Hardware-based address translation: transforms each virtual address into a physical address.

Assumptions

• Simplified model: address space contiguous in physical memory, equal-sized, smaller than physical memory.
• Unrealistic, but helps introduce basic mechanisms.

Example

• Process thinks memory starts at 0 and extends up to its maximum (e.g., 16KB).
• OS relocates process to another physical address (e.g., 32KB).
• Hardware must translate virtual addresses to physical addresses to maintain the illusion.

Dynamic Relocation (Base and Bounds)

• Two hardware registers: base and bounds (limit).
• Base register: added to virtual address to form physical address.
• Bounds register: ensures address is valid; violations raise exceptions.
• OS sets base/bounds for each process; saved/restored on context switches.
• Provides protection: processes cannot access memory outside their space.

Hardware Support

• CPU modes: kernel mode (can change base/bounds), user mode (cannot).
• Exceptions: raised for out-of-bounds memory access or privileged instruction attempts.
• Free list: OS tracks unused memory slots.

Operating System Responsibilities

• Allocate memory for processes, reclaim it when they exit.
• Manage base/bounds during context switches.
• Handle exceptions by terminating misbehaving processes.
• Can move processes by copying memory and updating base register.

Downsides

• Internal fragmentation: fixed slot allocations waste memory if stack/heap are small.
• Leads to more advanced mechanisms (segmentation, paging).

Summary

Address translation enables virtual memory by mapping virtual addresses to physical addresses. Base and bounds registers ensure transparency, efficiency, and protection. The OS collaborates with hardware: setting registers, handling exceptions, and managing free memory. Though effective, this approach wastes space (internal fragmentation), motivating more advanced designs.