Graduate Operating Systems
(Memory Management)

Fall 2020

Paper "Superpages"

- Small pages vs. large pages
- TLB coverage (how large should it be?)
- Hardware-imposed constraints
  - Page sizes supported by hardware
  - Contiguous in physical and virtual address space
  - Starting address must be multiple of its size
  - TLB uses single set of reference/dirty/protection bits for page

Paper "Superpages"

- Relocation-based allocation
  - When is relocation needed?
- Reservation-based allocation
  - What is the problem with this approach?
- Fragmentation control: "contiguity as a resource"
- Promotion
  - Challenges: who/when to promote
- Demotion
  - Challenge: how do we know which ":sub pages" are used
- Eviction
  - Challenge: dirty bits

Paper "Superpages"

- Proposed solution: reservation-based approach
- Buddy allocator
- Multi-list reservation scheme
- How to choose superpage size?
  - Dynamically-sized objects
  - Fixed-size objects

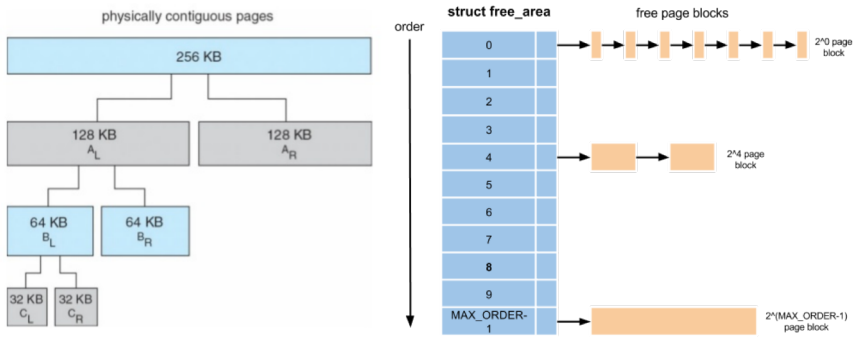
Paper “Superpages”

- Preempting reserved (unused) frames
- Coalescing of available memory regions
  - Contiguity-aware page replacement (active/inactive/cache lists)
- Incremental promotions
  - Cascading promotions possible
- Speculative demotions
  - E.g., due to eviction; to next-smaller size
  - Probabilistic demotions

Paper “Superpages”

- Dirty superpages
  - Demote clean superpages when writing occurs
- Population map
  - Keeps track of allocated base pages
- Wired page clustering

Paper “Superpages”



Paper “Superpages”

- Incremental promotions
- Speculative demotions
- Dirty superpages
- Reservation lists
- Population map
- Contiguity-aware page daemon
- Wired page clustering