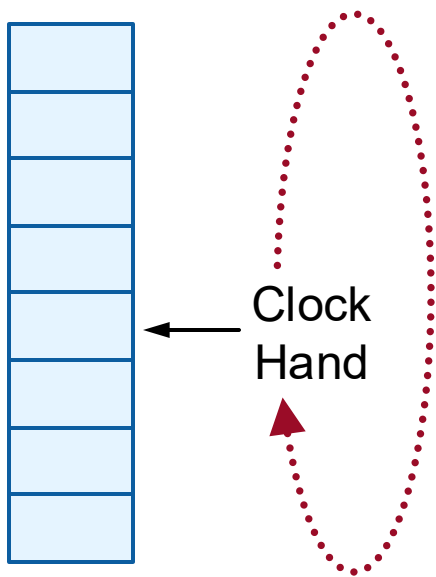


CS 111 Assignment 6: Page Replacement with the Clock Algorithm



The Clock Algorithm

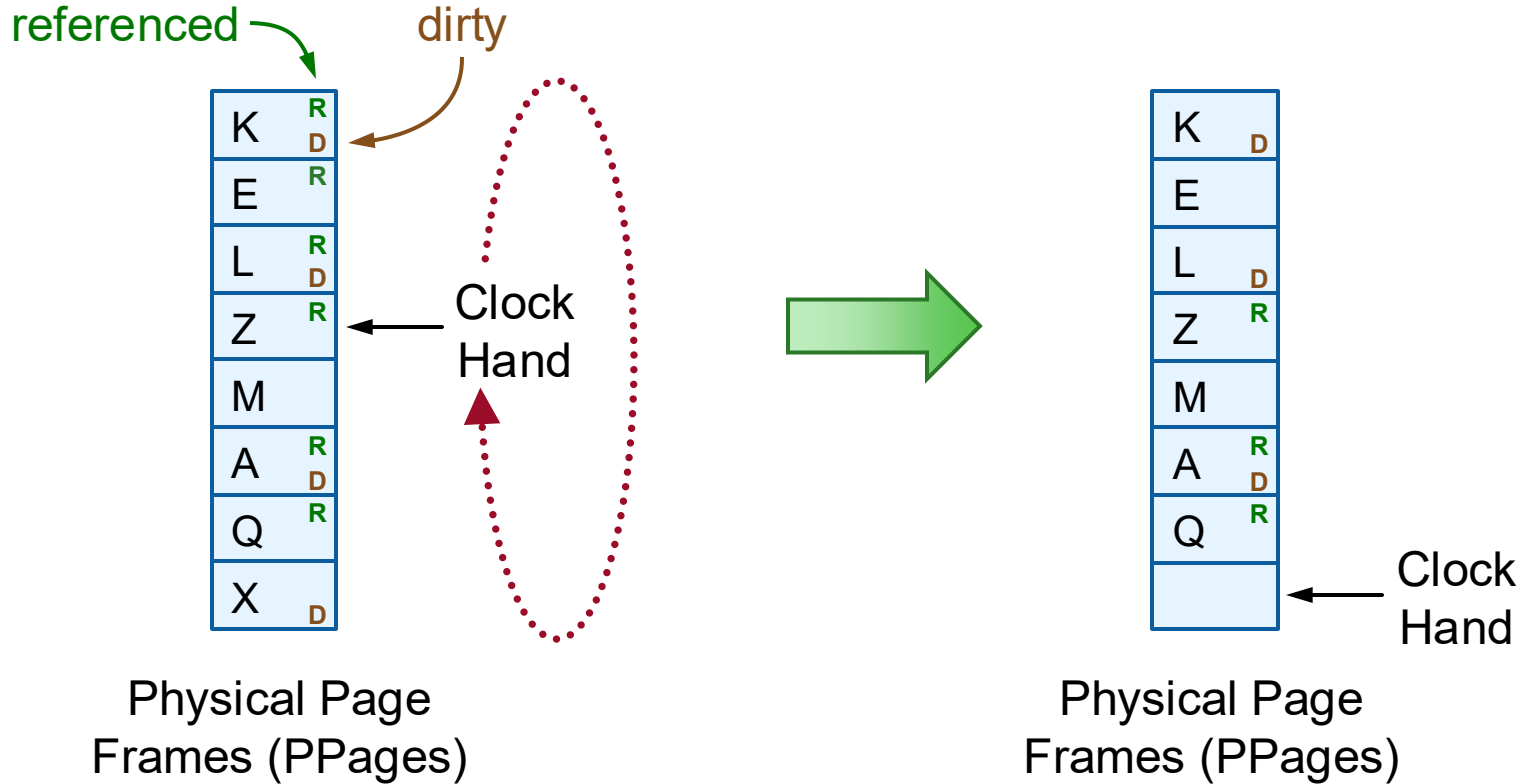


Physical Page
Frames (PPages)

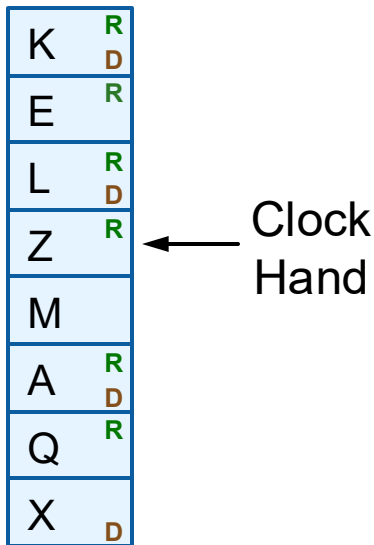
If need PPage but all in use:

- **Advance hand to next page, check referenced**
- **Referenced?**
 - Clear referenced
 - Try next page
- **Not referenced? Evict:**
 - Write back to file if dirty
 - Unmap from VMRegion
- **Then use PPage for new VPage**

Clock Example



Information Organization



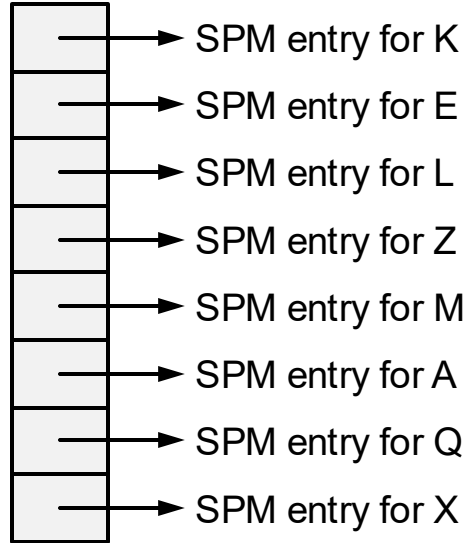
Physical Page
Frames (PPages)

- **How do we get to the data we need?**
 - Clock algorithm scans PPages
 - But dirty and reference bits are in the supplemental page map
 - Need MCrptFile to write back data, unmap page
- **Need a way to find VPage info for each PPage**

Reverse Page Map

| | |
|---|--------|
| K | R D |
| E | R |
| L | R D |
| Z | R |
| M | |
| A | R D |
| Q | R |
| X | D |

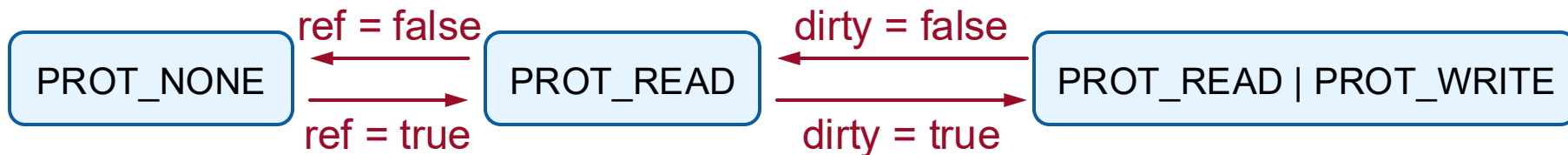
Physical Page
Frames (PPages)



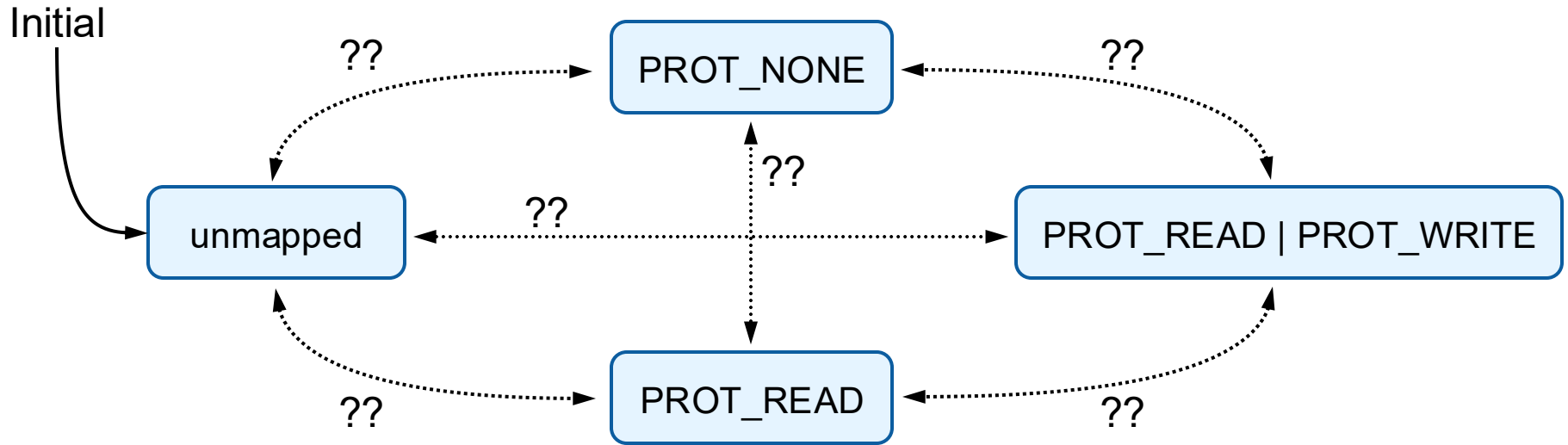
- **How to find reverse page map entry for a PPage, and vice versa?**
 - PPages are consecutive in memory, starting at `PhysMem::pool_base()`
 - `PhysMem::npages()` returns the total number of PPages
- **Be sure to update the reverse page map whenever PPages are allocated/freed!!!**

How to Compute Referenced?

- No access to hardware “referenced” bits for pages; must compute
- Similar approach to computing “dirty”

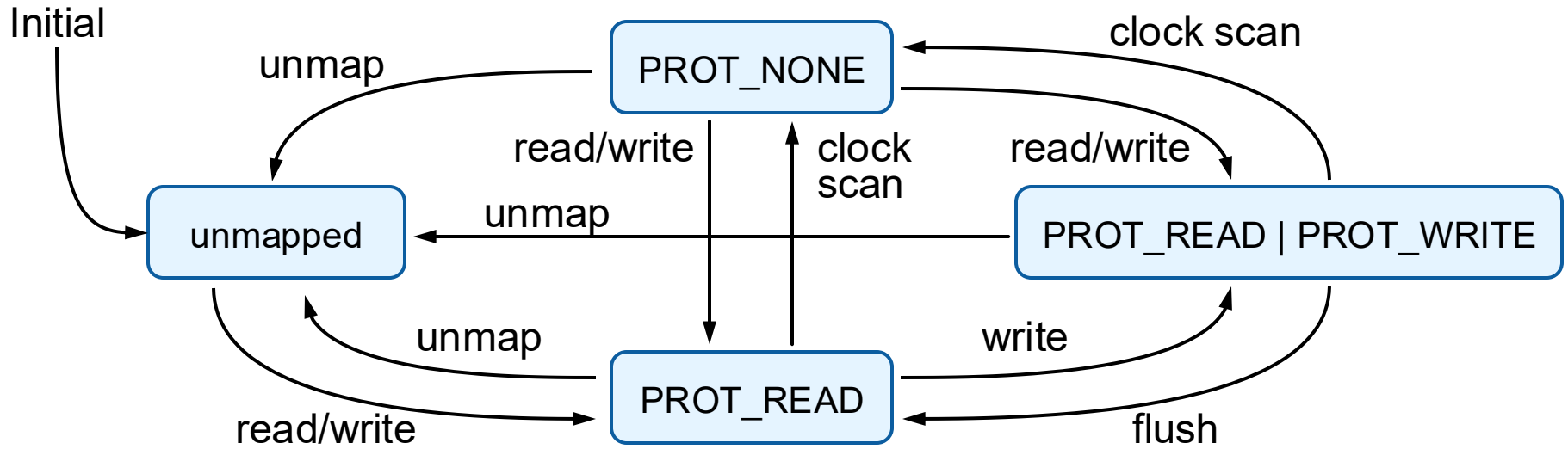


State of a VPage



- Which transitions are possible?
- When do they happen?
- When are dirty/referenced bits set/cleared?

State of a VPage



In which states can the dirty bit be set?

Consistency

- **Three structures:**
 - Supplemental page table
 - Hardware page table (invisible)
 - Reverse page map
- **Must make sure these are always consistent!! For example:**
 - Call `VMRegion::map()` whenever protections changed in SPM
 - Update reverse page map whenever `PhysMem::page_alloc()` or `PhysMem::page_free()` is called

Coding Tip

- Don't use [] with unordered_maps!!

```
std::unordered_map<int, Foo> map;  
Foo f = map[24];
```

- [] does auto-insert:

- If the desired entry doesn't exist, a new one will be created
- Usually isn't the desired behavior (e.g. unmapped page may appear to be mapped)

- Instead, use iterators:

```
auto it = map.find(24);  
if (it != map.end()) {  
    ...  
}
```