CISC 3320
C20b Thrashing & Working-Set Model

Hui Chen

Department of Computer & Information Science

CUNY Brooklyn College

Acknowledgement

- These slides are a revision of the slides provided by the authors of the textbook via the publisher of the textbook

Outline

- Thrashing

- Memory-Mapped Files
- Allocating Kernel Memory
- Other Considerations
- Operating-System Examples

Thrashing

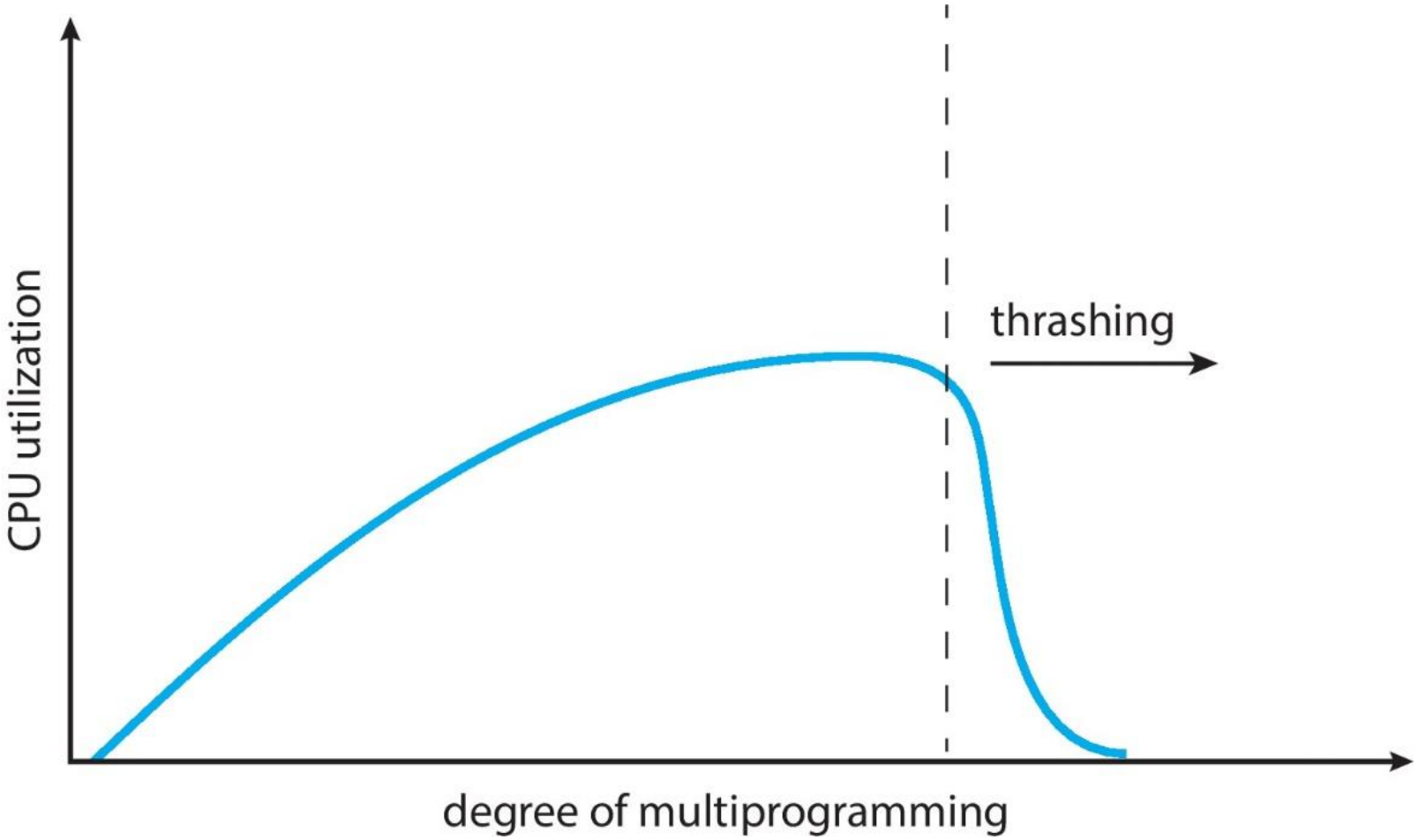
- A process is busy swapping pages in and out
- When does it happen?

Thrashing: When Does it Happen?

- Example:
  - If a process does not have "enough" pages, the page-fault rate is very high
- Page fault to get page
- Replace existing frame
- But quickly need replaced frame back

Thrashing: Problem

- This leads to:
  - Low CPU utilization
  - Operating system thinking that it needs to increase the degree of multiprogramming
  - Another process added to the system …


CPU utilization
thrashing
degree of multiprogramming

Demand Paging and Thrashing

- Why does demand paging work?
  - **Locality model**
  - Process migrates from one locality to another
  - Localities may overlap
- Why does thrashing occur when demand paging is in use?

  Σ size of locality > total memory size

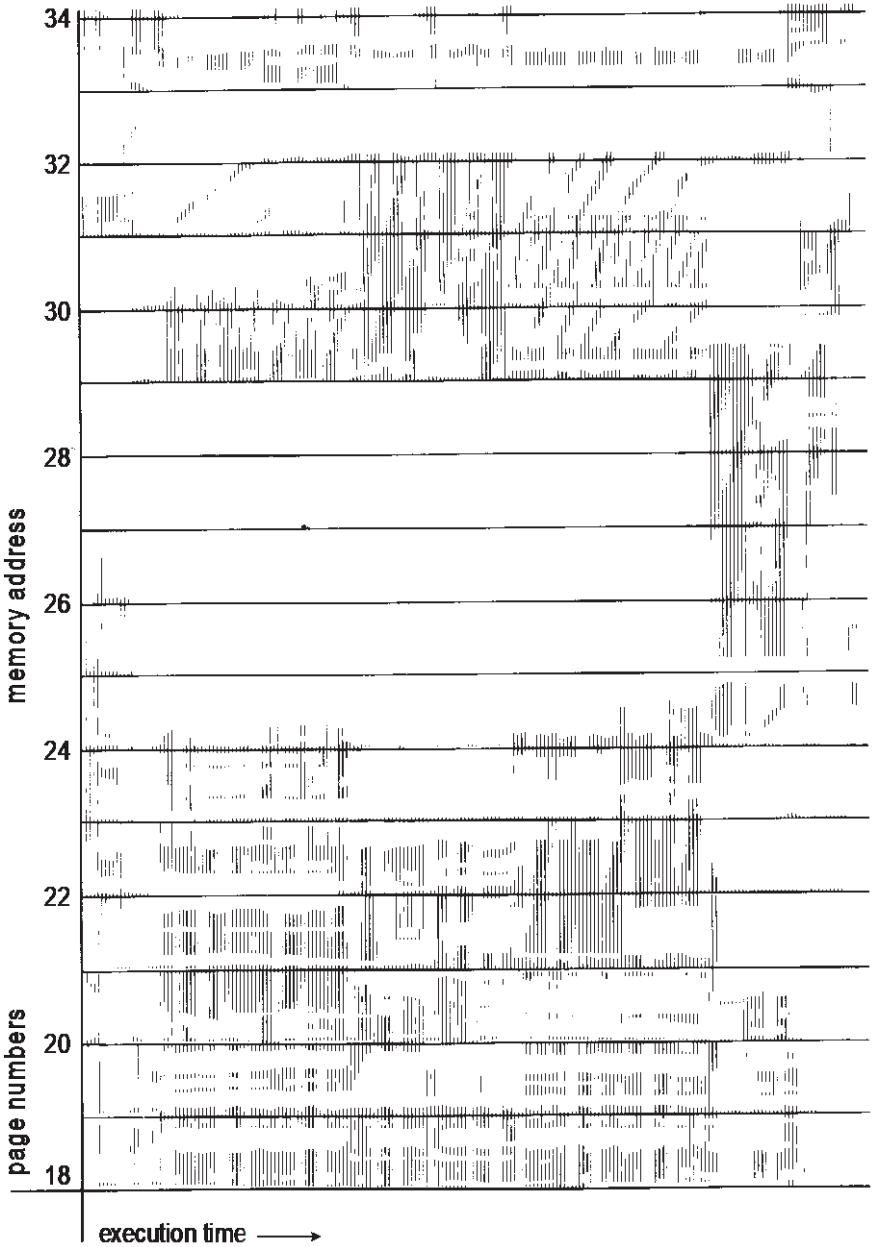
- Limit effects by using local or priority page replacement

Tackling Trashing

- Working-set model
- Page-fault frequency

Locality In A Memory-Reference Pattern

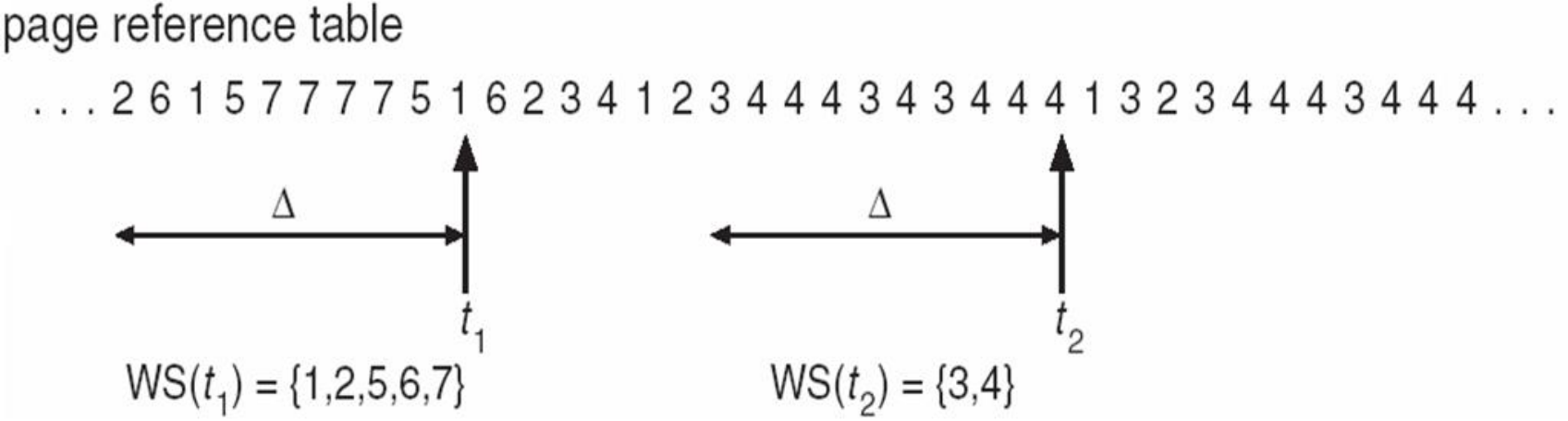
- Example


34
32
30
28
26
24
22
20
18
memory address
page numbers
execution time

Working-Set Model

- $\Delta \equiv$ working-set window $\equiv$ a fixed number of page references
  Example: 10,000 instructions
- WSS_i (working set of Process P_i) = total number of pages referenced in the most recent Δ (varies in time)
  - if Δ too small will not encompass entire locality
  - if Δ too large will encompass several localities
  - if $\Delta = \infty \Rightarrow$ will encompass entire program
- $D = \Sigma\ WSS_i \equiv$ total demand frames
  - Approximation of locality

Example


page reference table
. . . 2 6 1 5 7 7 7 7 5 1 6 2 3 4 1 2 3 4 4 4 3 4 3 4 4 4 1 3 2 3 4 4 4 3 4 4 4 . . .
Δ
t_1
Δ
t_2
WS(t_1) = {1,2,5,6,7}
WS(t_2) = {3,4}

Working-Set Model and Thrashing

- if $D > m \Rightarrow$ Thrashing

Reducing Thrashing

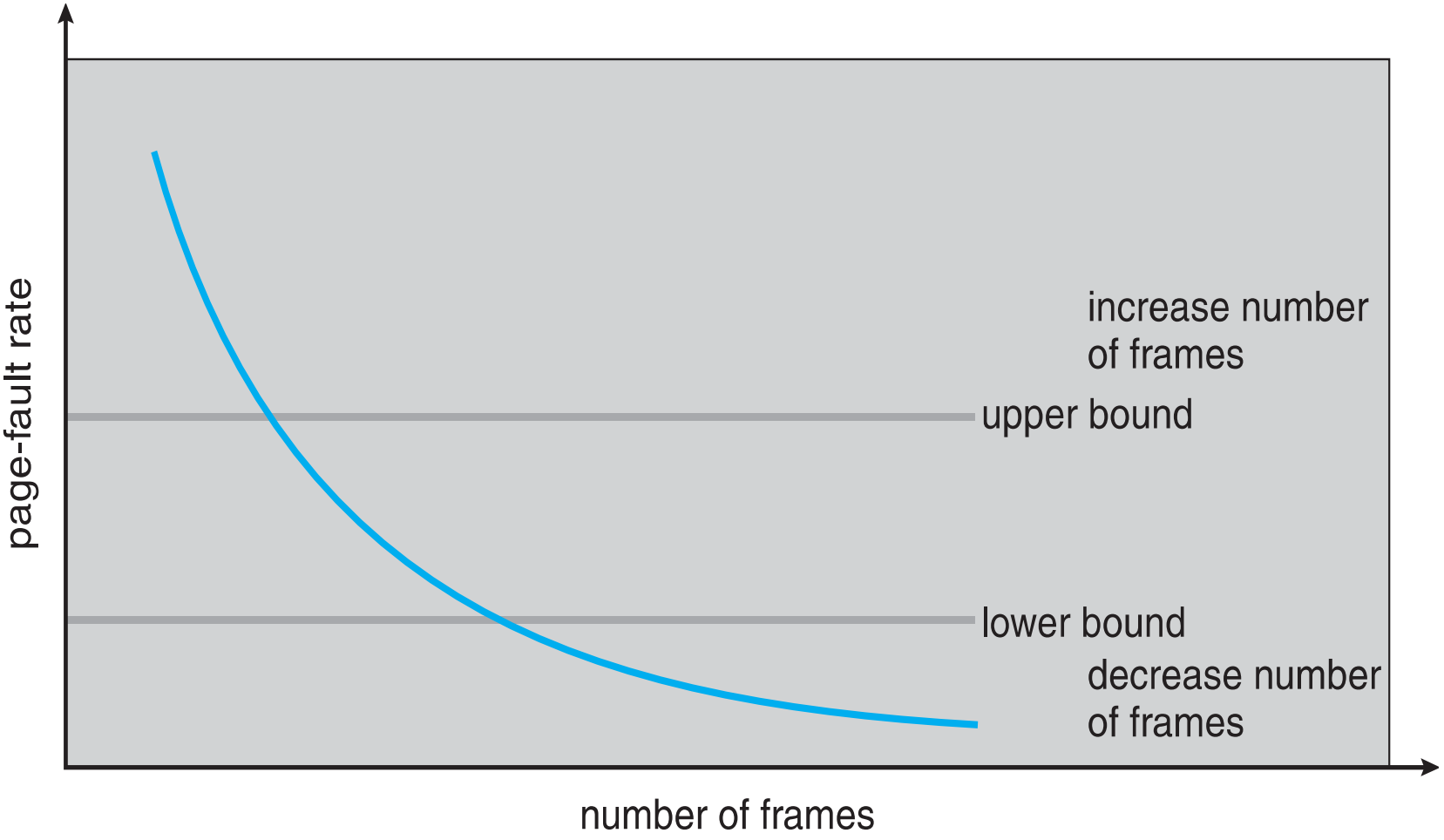
- Policy if $D > m$, then suspend or swap out one of the processes

Keeping Track of the Working Set

- Approximate with interval timer + a reference bit
- Example: Δ = 10,000
  - Timer interrupts after every 5000 time units
  - Keep in memory 2 bits for each page
  - Whenever a timer interrupts copy and sets the values of all reference bits to 0
  - If one of the bits in memory = 1 $\Rightarrow$ page in working set
- Why is this not completely accurate?
- Improvement = 10 bits and interrupt every 1000 time units

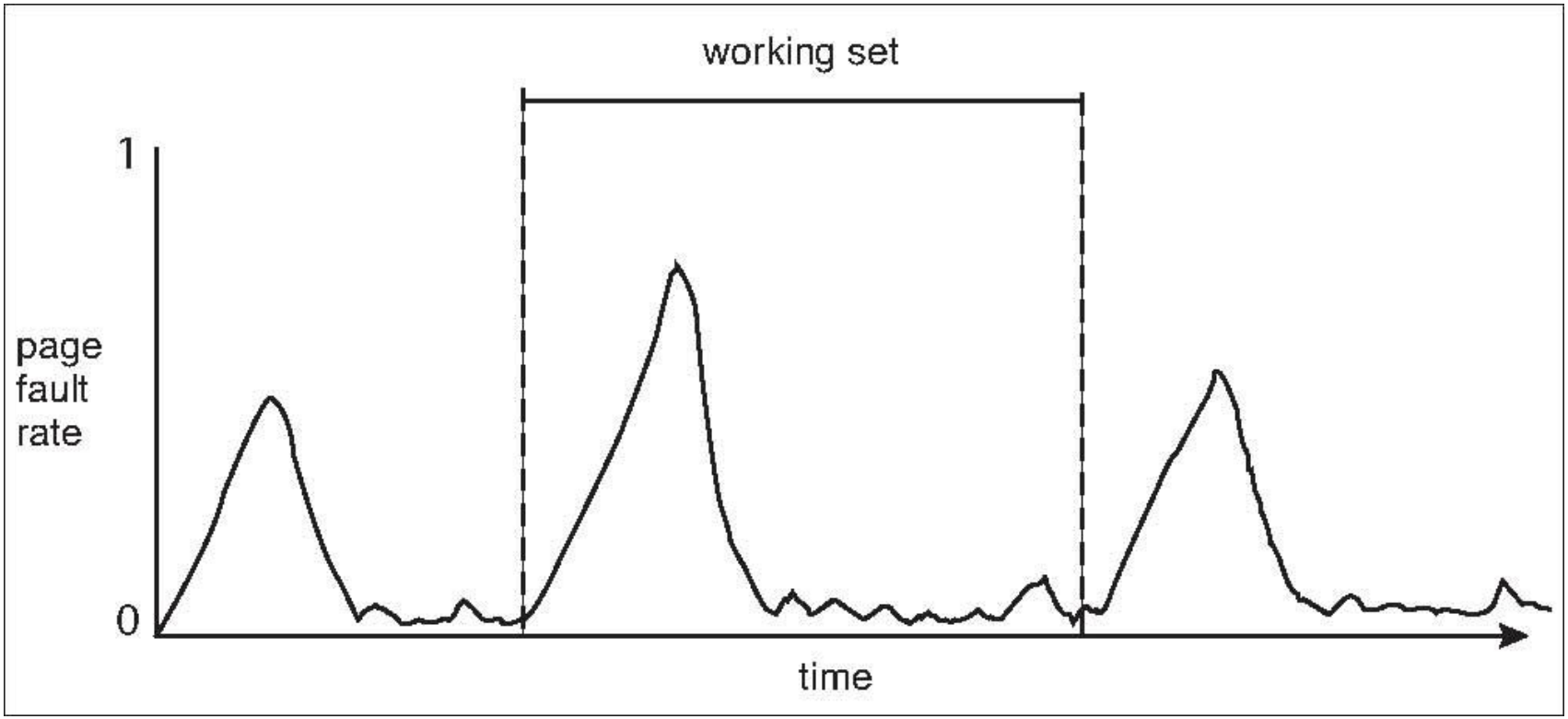
Page-Fault Frequency

- More direct approach than WSS
- Establish "acceptable" **page-fault frequency** (**PFF**) rate and use local replacement policy
  - If actual rate too low, process loses frame
  - If actual rate too high, process gains frame


page-fault rate
increase number
of frames
upper bound
lower bound
decrease number
of frames
number of frames

Working-Set and Page Fault Rate

- Direct relationship between working set of a process and its page-fault rate
- Working set changes over time
- Peaks and valleys over time


working set
1
page
fault
rate
0
time

Questions?

- Trashing
- Working-set model
- Page fault frequency