Working set model

Δ ≡ working-set window ≡ 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 Δ = ∞ ⇒ will encompass entire program

$D = \Sigma\ WSS_i \equiv$ total demand frames

- Approximation of locality

WSS = 3

P1 = 100 pages

D > m ⇒ Thrashing

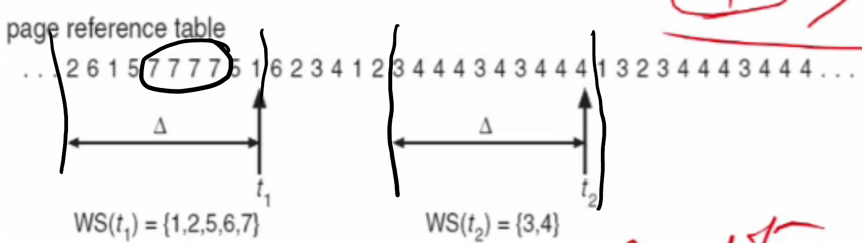
m = Total size of memory

Working-Set Model (Cont.)

if $D > m \Rightarrow$ Thrashing

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

D > m



D = Sum of size of locality

m = total size of memory

Typical usage of write policies

(1) Write back cache with write allocate
- In order to capture subsequent writes to the block in cache

(2) Write through cache with no-write-allocate
- since subsequent writes still have to go to the main memory

Virtual Memory

(1) Your system will support (install) very few softwares.

(2) Response time very high.
non volatile
→ Large memory where all the software's OS, games, video, music, etc