

OPERATING SYSTEM

Thrashing in Operating Systems

Definition

Thrashing is a condition in an operating system where the **system spends most of its time swapping pages between main memory (RAM) and disk instead of executing processes**.

In thrashing, the CPU remains mostly **idle**, even though many processes are present in the system.

Why Thrashing Occurs

Thrashing mainly occurs due to **insufficient physical memory** and **high multiprogramming level**.

Common causes include:

- Too many processes running at the same time
- Each process having **too few frames**
- Frequent **page faults**
- Poor memory management policies

What Happens During Thrashing

1. A process requests a page
2. Page is not in memory → **page fault**
3. OS loads page from disk
4. Another page is removed from memory
5. The removed page is soon needed again
6. Continuous page faults occur

System keeps **paging instead of computing**

Effects of Thrashing

- Very high paging activity
- Sharp drop in CPU utilization
- Poor system performance
- Increased disk I/O
- System becomes slow or unresponsive

Simple Example

Imagine:

- RAM can hold **10 pages**
- Each process needs **5 pages** to run properly
- OS loads **3 processes** (needs 15 pages)

- Memory is insufficient
- Pages are continuously swapped
- Thrashing occurs

Thrashing vs Normal Paging

Feature	Normal Paging	Thrashing
Page Fault Rate	Low	Very High
CPU Utilization	High	Very Low
System Performance	Good	Poor

How Thrashing Can Be Prevented

- Reduce degree of multiprogramming
- Use **Working Set Model**
- Use **Page Fault Frequency (PFF)** control
- Allocate sufficient frames to each process
- Use better page replacement algorithms

Thrashing is a condition in which an operating system spends most of its time handling page faults rather than executing processes, leading to poor performance and low CPU utilization.