Memory Management techniques and Processes Scheduling

Wael Alabdulaly

Memory management techniques, is the method responsible for managing the primary memory in computer memory management function keeps following of the current status in memory location, in case if it's free or allocated. It measure how memory is allocated over processes, deciding which gets memory, when they receive it, and how much they are free. Processes Scheduling simply is managing the processes residing in the main memory. To express the purpose of scheduling than we can say scheduling forma layout in which the already prioritize processes are loaded into the ready queue of the system and then send for the execution. The scheduling activity usually broken down into three different levels: short, medium, and long -term scheduling. In the following will discuss process, thread, and real-time Scheduling.

Different memory management techniques:

Six famous memory management techniques are: Fixed Partitioning, Dynamic Partitioning, Simple Paging, Simple Segmentation, Virtual-Memory Paging and Virtual-Memory Segmentation.

- Fixed partitioning: this partitioning approach divided into a fixed number of partitions just one process can be loaded into one partition at the same time. Strengths of this approach: easy to implement it and slandered method as a partitioning solution. Weaknesses of this approach insufficient use because of the internal fragmentation, must know the maximum number of active processes can run is fixed size of the task is limited to largest partition size, degree of multiprogramming limited by the number of partitions, memory is wasted in the partition, must translate relative address to physical address.

- Dynamic Partitioning: Partitions are created dynamically, each process loaded into a partition is exactly have same size as the process. Strengths of this approach are: ensure more efficient use of the main memory and no internal fragmentation. Weaknesses of this approach are inefficient use of processor because of the need for compaction and external fragmentation.

- Simple Paging: Strengths of this approach are: no need for external fragmentation, transfers between disks can be at the granularity of individual pages. Weaknesses of this approach are: maybe there is no correspondence between page protections. Settings and application data structures, requiring per process page tables, usually operating system need more storage for its internal data structures.

- Simple Segmentation: Strengths of this approach is no internal fragmentation. Weaknesses of this approach are: reduce the overhead compared to dynamic partitioning approach and improved the memory utilization

- Virtual-Memory Paging: Strengths of this approach are: having large virtual address space, no external fragmentation and higher degree of multi programming. Weaknesses of this approach is overhead coming from the complex of memory management.

- Virtual-Memory Segmentation: Strengths of this approach are: it supports a high level of multiprogramming especially the enormous virtual address space and no internal fragmentation. Weaknesses of this approach is the overhead of complex memory management.

The Use of Processes Scheduling:

- *Author name is WAEL AL-ABDULALY currently pursuing master degree in computer information systems in Florida institute of technology, Melbourne, FL, USA. E-mail: waelalabdulaly@gmail.com*

The use of processor scheduling is to select processes to execute by the processor over time, such that it meets the system objectives, throughput, and processor efficiency. Process scheduling is an essential part of a multiprogramming operating system Memory at a time and loaded process shares the CPU using time multiplexing. The process could create new sub process and will wait for its termination. The process could be removed forcibly from the CPU, as a result of interrupt and put back in the ready queue. Two State Process Model: running and non-running. Running is created by OS that process enters system as running. Not running that are not running stay in queue, waiting for their turn to execute. Each entry in the queue is a pointer to a particular process. Queue is implemented by using linked list. Use of dispatcher is as following: When a process is interrupted, the same process is transferred in the waiting queue. If the process completed the process is discarded. Process Scheduling Algorithms: First Come First Serve, Shortest Job First, Priority Scheduling, Round Robin (RR) Scheduling and Multilevel Queue Scheduling.

Thread:

an approach provide a way to improve application performance through parallelism also it is a flow of execution through the process code, with its own program counter, system registers and stack. Threads represent a software approach to improving performance of operating system by reducing the overhead thread that is equivalent to a classical process. Thread Scheduling: It is utilized in grained parallelism. Approaches to Thread Scheduling: Load Sharing, Dedicated Processor Assignment and Dynamic Scheduling

Real Time Scheduling:

arrange and manage the order of tasks executions, however there are two types of Real time Scheduling: Static priority scheduling and dynamic priority scheduling, RM (Rate Monotonic) Optimal static priority scheduling it assigns priority according to duration of task with less time has a higher priority executes a job with the less period. Optimal dynamic priority scheduling a task with less deadline has a higher priority executes a job with the earliest deadline.

References

Operating System Concept, 9/E, Silberschatez, Galvin, Gagne, Wiley.

Protic, J., Tomasevic, M., & Milutinovic, V. (1996). Distributed shared memory: Concepts and systems. IEEE Parallel Distrib. Technol., Syst. Appl. IEEE Parallel & Distributed Technology: Systems & Applications, 4(2), 63-71.

Operating Systems: Internals and Design Principles" by William Stallings