Tuijin Jishu/Journal of Propulsion Technology
ISSN: 1001-4055
Vol. 43 No. 4 (2022)

Memory Management and Virtual Memory Concept

[1] Ashesh Sharma, [2] Aakash Bhati, [3] Laxmi Jangir, [4] Aditaya Kumar Joshi

[1] Asst. Professor
Dept. of Management
Arya Institute of Engineering and Technology, Jaipur
[2] Asst. Professor
Civil Engineering
Arya Institute of Engineering Technology & Management, Jaipur
[3] Research Scholar
Artificial Intelligence & Data Science
Arya Institute of Engineering and Technology, Jaipur
[4] Research Scholar
Artificial Intelligence & Data Science
Arya Institute of Engineering and Technology, Jaipur

Abstract: This paper is crafted to simplify the intricate concepts of memory control and digital memory. It starts by examining essential elements, including the memory hierarchy, the pivotal role of the Memory Management Unit (MMU), and various addressing modes. Delving deeper, the paper explores memory allocation techniques such as contiguous and non-contiguous methods. Contiguous allocation allocates memory blocks in a sequential manner, offering straightforward access but potentially leading to fragmentation issues. In contrast, non-contiguous allocation utilizes techniques like paging or segmentation, ensuring efficient use of memory space but requiring complex management strategies. The paper discusses the advantages, challenges, and implementation details associated with both techniques, providing readers with a nuanced understanding. By elucidating these concepts concisely, the paper empowers readers to grasp the complexities of memory control in digital systems. Its structured approach ensures that readers, whether beginners or enthusiasts, can comprehend the fundamental principles governing memory management. With a blend of theoretical insights and practical considerations, this paper serves as a valuable resource, offering a clear and insightful overview of memory control and digital memory concepts, fostering a deeper understanding of these pivotal aspects in computer science and technology.

Keywords: Operating system, Virtual memory, Memory Management, Demand paging, Paging, Virtual storage.

1. Introduction

Operating system (OS) is system software that manages the coordination between computer hardware and software and provides common services for computer programs. All computer programs except firmware require an operating system to function. Memory is the most important part of a computer system that provides a collection of data in a specific format which stores the instructions and process of data. The memory comprises a large group of words or bytes, each with its location. For any given system its primary purpose is to execute programs, which along with the information they access, should be stored in the main memory during execution.

Memory Management Techniques

The memory management techniques can be classified into the following main categories:

Contiguous memory management scheme. In this scheme, each program occupies a single contiguous block of storage location i.e., a set of memory locations with consecutive addresses. It is the simplest memory management scheme used in the earliest generation of computer systems. In which the main memory is divided into two contiguous areas or partitions. The operating systems reside permanently in one partition, generally at the lower memory, and the user process is loaded into the other partition. The advantages of this are as Simple to implement, easy to manage and design, and in a single contiguous memory management scheme, once a process is loaded it is given full processor time, and no other processor will interrupt it. On the other hand, Disadvantages include wastage of memory space due to unused memory as the process is unlikely to use all the available memory space, does not support multiprogramming, it cannot be executed if the program is too large the binary image into the main memory, CPU remains idle waiting for the disk to load the binary image into the main memory.

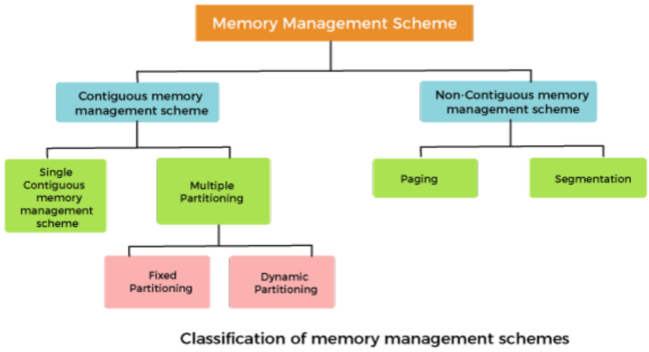

Fig 1: Classification of memory management schemes

Non-contiguous memory management scheme. In a Non-Contiguous memory management scheme, the program is divided into different blocks and loaded at different portions of the memory that need not necessarily be adjacent to one another.

Paging: It is a technique that eliminates the need for contiguous allocation of main memory. The main memory is divided into fixed-size blocks of physical memory called frames. The size of a frame should be kept the same as that of a page to maximize the main memory and avoid external fragmentation

2. Virtual memory

Virtual memory is a laptop system enhancement that extends the available bodily reminiscence (RAM) to optimize performance and enable multitasking. It does this by using stimulating additional memory the use of a mixture of RAM and Hard disk area. This approach allows larger and extra complicated programs to run effectively. To run applications requires a part of physical reminiscence, the operating system transfers less frequently used records from RAM to a chosen area at the tough drive, we name it a paging file. This frees up RAM for active duties even as transferring statistics inside and outside of digital reminiscence. The concept of virtual memory has diverse advantages. It permits more packages and information to be processed simultaneously, improving the overall performance of a computer.

It offers an efficient technique of memory control. But if you depend an excessive amount of on digital memory, your machine might also lag the reason is that tough drives are slower than RAM. To maintain the right virtual reminiscence operation, the running system has to balance the data transfer between RAM and the difficult disc cautiously. Algorithms that determine which information must be saved in RAM for a brief get right of entry and which need to be temporarily transferred to digital memory are used to acquire this. Since those algorithms have an impact on how users engage with apps and multitask, they're critical for preserving a machine's overall performance and responsiveness. The running system should manage facts glide among random get entry to reminiscence (RAM) and the hard disc to provide optimal virtual memory overall performance successfully. Algorithms that predict which records must be quickly moved to virtual memory and which must be kept in RAM for clean get right of entry are used to achieve this. This technique immediately impacts how customers engage with applications and multitask, as a result, it's essential to maintain the highest viable device performance and responsiveness.

The working machine ought to expertly manipulate the switch of statistics between RAM and the difficult disc to hold effective virtual memory operation. This is accomplished with the aid of the usage of algorithms that decide which statistics have to be temporarily moved to virtual reminiscence and which have to be stored in RAM for clean get admission. These algorithms are vital for retaining system responsiveness and overall performance because they affect how the person reports the usage of programs and deals with several duties right now.

Demand paging

Demand paging is a reminiscence management method used in modern-day laptop working structures. It allows strategies to use virtual reminiscence efficaciously and optimizes bodily reminiscence utilization with the aid of loading the important pages into memory, with additional pages added in as needed. The call for paging technique may be damaged down into several steps:

Virtual Memory Setup: Each system is allotted a digital cope with the area, which is larger than the real bodily memory available inside the gadget. This virtual cope with space is split into fixed-length blocks referred to as pages. The bodily reminiscence is divided into blocks referred to as frames, which can be normally equal in length to pages. The aim is to map pages from the virtual address space to frames in bodily reminiscence.

1. Page Table Initialization: Each manner has a web page desk that keeps music of the mapping between virtual pages and physical frames. Initially, the page table is installed to indicate that no pages are in physical reminiscence (i.e., all pages are marked as "not gift").
2. Process Execution: When a system starts, it starts executing its code. As it accesses unique parts of its virtual address area, it can try and get admission to pages that are not presently in bodily reminiscence.
3. Page Fault: When a technique tries to access a web page that is not in bodily memory, a web page fault happens. This is an exception that is raised by using the Memory Management Unit (MMU).
4. Handling Page Fault: When a web page fault occurs, the operating system wishes to reply. The steps involved in managing a web page fault are as follows:
5. Page Table Lookup: The running device consults the page table for the procedure to determine which virtual page caused the fault.
6. Secondary Storage Access: The OS identifies the region of the specified page on secondary storage (commonly a difficult force or SSD).
7. Page Loading: The page is then loaded from the secondary garage into a to-be-had body in bodily memory. If there are not any free frames, a web page alternative set of rules is used to select a page to evict from physical reminiscence. The evicted web page is written lower back to the secondary garage if it is changed into modified.
8. Update Page Table: The web page desk access for the accessed digital page is up to date to reflect that it is now found in physical memory.
9. Resuming Process Execution: Once the required page is loaded into physical memory, the system can resume its execution as though there was no interruption.
10. Lazy Loading: Demand paging commonly employs a "lazy loading" method, which means that not all pages are loaded into physical memory whilst the procedure begins. Instead, the best essential pages (e.g., code and records wished for the initial execution) are loaded. As the method maintains running and accessing its memory area, extra pages are loaded on-demand.
11. Page Replacement: If all physical memory frames are occupied while a web page fault occurs, the working machine needs to select a page to evict. This is performed with the use of a web page substitute algorithm, along with Least Recently Used (LRU) or Second Chance. The selected web page is swapped out to make room for the newly demanded web page.

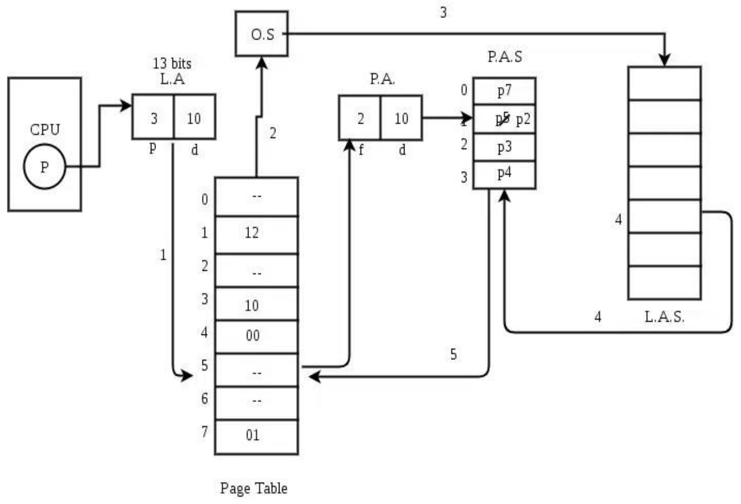

Fig 2: Steps of ready state

Optimization: Modern running structures employ various optimization strategies to decrease the impact of page faults and enhance overall performance. These include the usage of Translation Lookaside Buffers (TLB) for faster dealing with translation and clever web page replacement algorithms that goal to evict the least valuable pages.

3. Literature review

Historical Overview: Memory management has visible large advancements since the early days of computing. Early systems used fixed memory partitions, proscribing flexibility, and efficiency. The introduction of dynamic memory allocation, brought by the development of the LISP programming language, marked a pivotal moment. Virtual memory was first conceptualized in the Nineteen Sixties, paving the way for more sizable use of the secondary garage as an extension of bodily memory. Virtual Memory Systems: Virtual memory is a key innovation that lets in procedures to address more memory than is physically to be had. The paging and segmentation techniques, delivered in the Nineteen Seventies, have enabled efficient memory allocation, safety, and sharing among techniques. These principles have advanced through the years, and modern digital memory structures, which include the ones primarily based on demand paging and using translation look-aside buffers (TLB), have turned out to be indispensable components of working structures.

Memory Management Algorithms: Memory management algorithms play an essential position in optimizing the allocation and deallocation of reminiscence resources. Classical algorithms like First-Fit, Best-Fit, and Worst-Fit were broadly studied, with more recent algorithms like the Buddy System and Slab Allocation presenting extra green answers. Research maintains to recognition of growing adaptive memory allocation techniques which could dynamically regulate varying workloads and useful resource availability.

Memory Virtualization for Cloud Computing: Cloud computing has introduced new challenges to reminiscence control. Virtualization technologies, together with hypervisors, permit the efficient sharing and management of reminiscence resources amongst virtual machines (VMs). Techniques like reminiscence ballooning and transparent page sharing have been employed to optimize memory utilization and reduce overhead in virtualized environments.

Security and Memory Management: Security concerns associated with reminiscence control have won prominence. Memory-based total attacks, along with buffer overflows and reminiscence corruption, have triggered research into strategies like address area format randomization (ASLR) and records execution prevention (DEP) to defend against malicious exploits.

Future Challenges and Trends: As computing environments continue to conform, new challenges and traits in memory control and virtual memory are rising. These encompass the adaptation of reminiscence structures for non-volatile memory (NVM) technology, addressing the memory requirements of AI and big data workloads, enhancing strength performance, and ensuring strong reminiscence control for actual-time and edge computing packages.

Conclusion: Memory management and digital memory concepts have come in a protracted manner because of their inception. They are crucial components of contemporary computing structures, enabling green useful resource allocation and continuing consumer enjoyment. As the era advances, it's miles imperative to hold getting to know and growing novel memory control techniques to address the evolving demands of computing environments, while additionally thinking about security, performance, and adaptability in an ever-changing digital panorama. This literature review gives insights into the wealthy history and promising future of reminiscence control and digital reminiscence principles.

4. Future Scope

Memory Optimization for AI and Machine Learning: With the growing significance of synthetic intelligence and system getting to know, memory management structures want to be optimized to effectively handle the reminiscence requirements of deep mastering fashions and large datasets. Future traits ought to attention to creating specialized memory control techniques to enhance the overall performance of AI workloads. Quantum Computing Memory Management: As the quantum computing era advances, conventional memory management strategies may not be sufficient. Future studies could discover a way to adapt reminiscence control for quantum computers, considering their precise residences and memory requirements. Hybrid Memory Systems: Destiny may additionally see greater substantial adoption of hybrid memory systems that combine various reminiscence technologies like DRAM, NAND flash, and rising non-volatile reminiscence technologies. Effective memory management strategies may be essential to leverage the benefits of those hybrid systems.

Memory Security: As protection threats keep conforming, there is a want for reminiscence management systems that could defend against memory-related vulnerabilities like buffer overflows and other styles of reminiscence corruption attacks. Future research may consciousness on growing memory control techniques that decorate security. Real-time Memory Management: Real-time systems, together with the ones used in self-reliant vehicles and critical infrastructure, require green reminiscence management to make sure well timed and predictable responses. Future work on this place may additionally concentrate on developing reminiscence control strategies that meet the stringent timing necessities of real-time structures. Memory Management for Edge Computing: With the proliferation of facet computing devices, reminiscence control will become extra essential. Edge devices often have confined sources, and efficient reminiscence control is crucial to ensure the most advantageous overall performance and responsiveness. Energy-Efficient Memory Management: Energy performance is a developing issue, specifically in cellular and IoT gadgets. Future studies should attention on developing memory management strategies that lessen power intake at the same time as maintaining overall performance.

Memory Management for Extended Reality (XR) and Gaming: Virtual fact (VR), augmented truth (AR), and gaming programs have disturbing memory necessities. Future memory control developments may additionally need to cater to these immersive and resource-intensive stories. Memory Management for Big Data and Cloud Computing: With the ever-growing volume of statistics being processed in cloud computing environments, efficient reminiscence control is critical. Future paintings might also deal with optimizing memory utilization in distributed structures and large statistics frameworks. Memory Management for Non-risky Memory (NVM): As non-unstable reminiscence technology like 3D Point and Intel Optane become greater widely widespread, reminiscence management desires to evolve to completely make use of the blessings of NVM in phrases of pace and patience.

5. Result and conclusion

Methods of Managing Memory: Every program in the contiguous memory management scheme uses a single contiguous block of memory. It's straightforward but ineffective because it wastes memory and doesn't support several programs. Non-contiguous memory management scheme: This type of memory management removes the necessity for contiguous allocation by dividing programs into distinct blocks and loading them at various memory addresses.

Paging: This method aligns equal-sized logical units (pages) with fixed-size blocks (frames) in the main memory. This reduces the fragmentation of memory.

Virtual Storage: RAM and hard drive space are combined to create virtual memory, which increases the amount of physical memory that is available. It provides a running large program, effective management of memory, and multitasking. Data transmission between RAM and the hard drive is handled by the operating system, although over-reliance on Request Paging: Demand paging is a technique that loads pages into memory when a page failure occurs. Interrupt handling, locating the needed page, loading it into physical memory, updating the page table, and starting the program again are all steps in the process. Page replacement algorithms are employed to determine which pages in any physical memory need to be replaced.

In summary for smooth computer operation effective memory management is essential. To maximize memory utilization and facilitate multitasking nowadays several memory management strategies have been developed. Through the use of RAM and hard disk space, virtual memory increases the amount of physical memory that is available, thereby enhancing overall performance. Demand paging, a crucial feature of virtual memory, guarantees that the necessary information is accessible in physical memory when needed, which is necessary for effectiveness and responsiveness.

References

[1] W. Appel and K. Li. Virtual memory primitives for user programs. In Proc. of 4th International Conference on Architectural Support for Programming Languages and Operating Systems, April 1991.

[2] J. Bartlett, J. Gray, and B. Horst. Fault tolerance in Tandem computer systems. In A. Avizienis, H. Kopetz, and J.C. Laprie, editors, The Evolution of Fault-Tolerant Computing, Springer Verlag, 1987.

[3] K.P. Birman. Replication and fault-tolerance in the ISIS system. In Proc. of 10th ACM Symposium on Operating Systems Principles, Washington, December 1985.

[4] H. E. Bal, M. F. Kaashoek, and A. S. Tanenbaum. Experience with distributed programming in Orca. IEEE Transactions on Software Engineering, March 1992.

[5] L. Brown. Fault Tolerant Distributed Shared Memories. PhD thesis, Florida Atlantic University, December 1993.

[6] L. Brown and J. Wu. Dynamic snooping in a fault-tolerant distributed shared memory. In Proc. of 14th International Conference on Distributed Computing Systems, Poznan, Poland, June 1994.

[7] L. Brown and J. Wu. Fault-tolerant distributed shared memories. The Journal of Systems and Software, May 1995.

[8] B. N. Bershad and M. J. Zekauskas. Midway: Shared memory parallel programming with entry consistency for distributed memory multiprocessors. Department of Computer Science, Carnegie-Mellon University, Pittsburgh, September 1991.

[9] J. B. Carter, J. K. Bennett, and W. Zwaenepoel. Implementation and performance of Munin. In Proc. of 13th ACM Symposium on Operating Systems Principles, {18 October 1991}.

[10] R. K. Kaushik Anjali and D. Sharma, "Analyzing the Effect of Partial Shading on Performance of Grid Connected Solar PV System", *2018 3rd International Conference and Workshops on Recent Advances and Innovations in Engineering (ICRAIE)*, pp. 1-4, 2018.

[11] R. Kaushik, O. P. Mahela, P. K. Bhatt, B. Khan, S. Padmanaban and F. Blaabjerg, "A Hybrid Algorithm for Recognition of Power Quality Disturbances," in *IEEE Access*, vol. 8, pp. 229184-229200, 2020.

[12] Kaushik, R. K. "Pragati. Analysis and Case Study of Power Transmission and Distribution." J Adv Res Power Electro Power Sys 7.2 (2020): 1-3.

[13] T. Manglani, A. Vaishnav, A. S. Solanki, and R. Kaushik, "Smart Agriculture Monitoring System Using Internet of Things (IoT)," 2022 International Conference on Electronics and Renewable Systems (ICEARS), Tuticorin, India, 2022, pp. 501-505.

[14] R. Kaushik et al., "Recognition of Islanding and Operational Events in Power System With Renewable Energy Penetration Using a Stockwell Transform-Based Method," in IEEE Systems Journal, vol. 16, no. 1, pp. 166-175, March 2022.

[15] Guru Saran Chayal, Bharat Bhushan Jain and Rajkumar Kaushik, "A Detailed Study of Electrical Vehicle with Improved Applications: A Review", International Journal of Engineering Trends and

Applications (IJETA), vol. 8, no. 6, pp. 31, Nov-Dec 2021.

[16] Sharma, R. and Kumar, G. (2017) "Availability improvement for the successive K-out-of-N machining system using standby with multiple working vacations," International journal of reliability and safety, 11(3/4), p. 256. doi: 10.1504/ijrs.2017.089710.

[17] Sharma, Richa and Kumar, Gireesh. "Availability Modelling of Cluster-Based System with Software Aging and Optional Rejuvenation Policy" Cybernetics and Information Technologies, vol.19, no.4, 2019, pp.90-100. https://doi.org/10.2478/cait-2019-0038