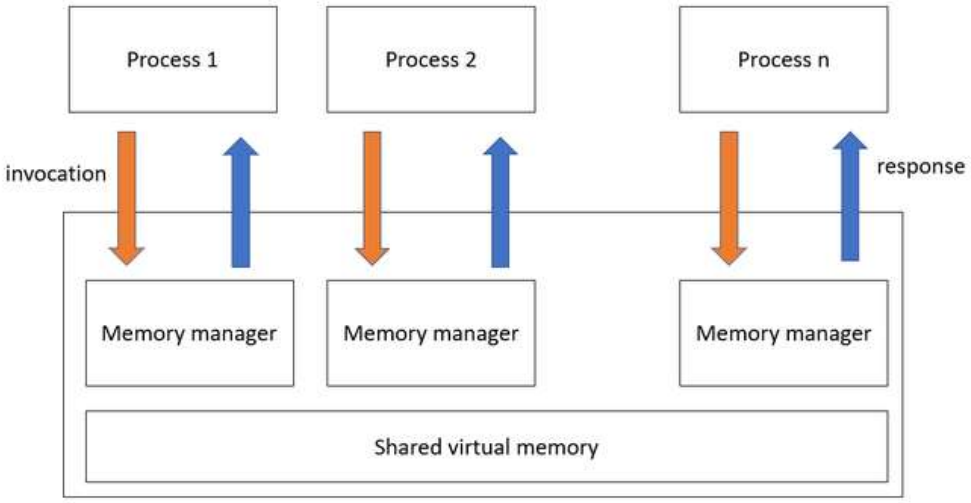
8 Distributed Shared Memory Systems and Programming

[Weightage(15%): Approx. 10-11 Marks out of 70 Marks]

- Distributed Shared Memory
- Implementing Distributed Shared Memory
- Distributed Shared Memory Programming Primitives

8.1 Distributed Shared Memory



Source: www.geeksforgeeks.org

The distributed shared memory (DSM) is a mechanism that manages memory across multiple nodes and makes inter-process communications transparent to end-users. The applications will think that they are running on shared memory. DSM is a mechanism of allowing user processes to access shared data without using inter-process communications. In DSM every node has its own memory and provides memory read and write services and it provides consistency protocols. The DSM implements the shared memory model in distributed systems but it doesn't have physical shared memory. All the nodes share the virtual address space provided by the shared memory model. The Data moves between the main memories of different nodes.

8.1 Distributed Shared Memory

Advantages of Distributed shared memory

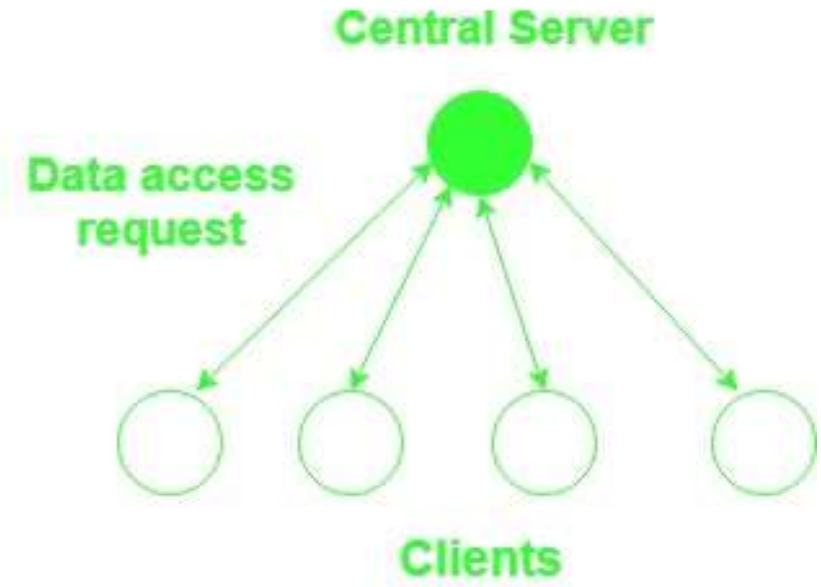
- **Simpler abstraction:** Programmer need not concern about data movement, As the address space is the same it is easier to implement than RPC.
- **Easier portability:** The access protocols used in DSM allow for a natural transition from sequential to distributed systems. DSM programs are portable as they use a common programming interface.
- **locality of data:** Data moved in large blocks i.e. data near to the current memory location that is being fetched, may be needed future so it will be also fetched.
- **on-demand data movement:** It provided by DSM will eliminate the data exchange phase.
- **larger memory space:** It provides large virtual memory space, the total memory size is the sum of the memory size of all the nodes, paging activities are reduced.
- **Better Performance:** DSM improve performance and efficiency by speeding up access to data.
- **Flexible communication environment:** They can join and leave DSM system without affecting the others as there is no need for sender and receiver to existing,
- **process migration simplified:** They all share the address space so one process can easily be moved to a different machine.

Source: www.geeksforgeeks.org

8.2 Implementing Distributed Shared Memory

Central Server Algorithm:

- In this, a central server maintains all shared data. It services read requests from other nodes by returning the data items to them and write requests by updating the data and returning acknowledgement messages.
- Time-out can be used in case of failed acknowledgement while sequence number can be used to avoid duplicate write requests.
- It is simpler to implement but the central server can become bottleneck and to overcome this shared data can be distributed among several servers. This distribution can be by address or by using a mapping function to locate the appropriate server.

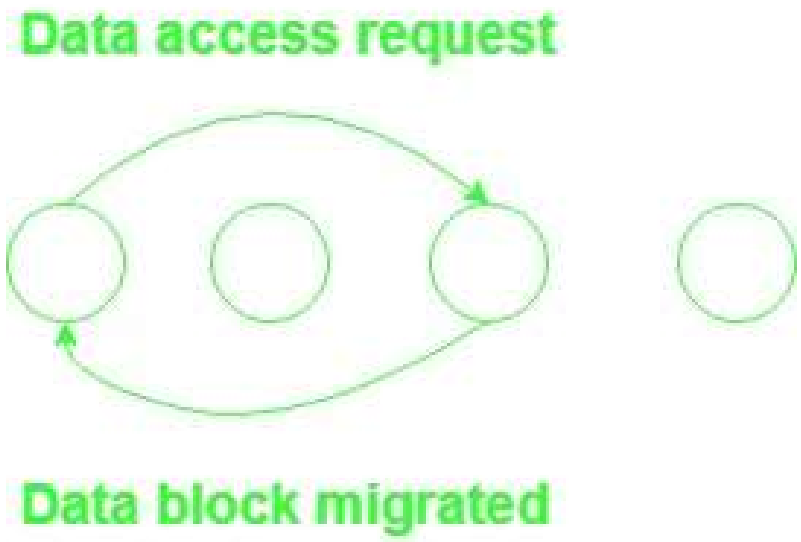


Source: www.geeksforgeeks.org

8.2 Implementing Distributed Shared Memory

Migration Algorithm:

- In contrast to central server algorithm where every data access request is forwarded to location of data while in this data is shipped to location of data access request which allows subsequent access to be performed locally.
- It allows only one node to access a shared data at a time and the whole block containing data item migrates instead of individual item requested.
- It is susceptible to thrashing where pages frequently migrate between nodes while servicing only a few requests.
- This algo provides an opportunity to integrate DSM with virtual memory provided by operating system at individual nodes.

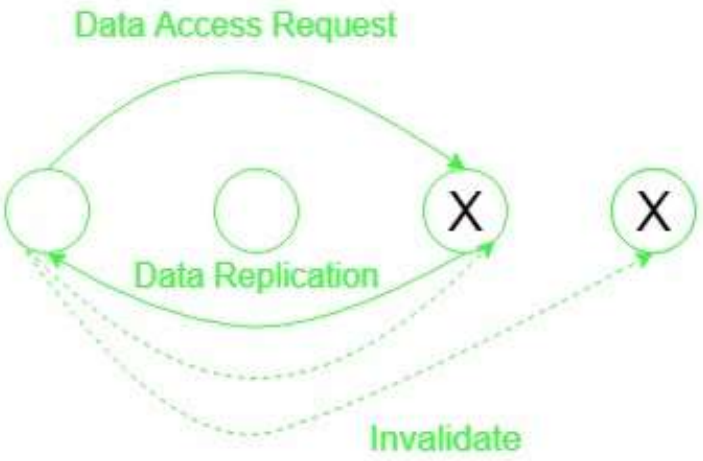


Source: www.geeksforgeeks.org

8.2 Implementing Distributed Shared Memory

Read Replication Algorithm:

- This extends the migration algorithm by replicating data blocks and allowing multiple nodes to have read access or one node to have both read write access.
- It improves system performance by allowing multiple nodes to access data concurrently.
- The write operation in this is expensive as all copies of a shared block at various nodes will either have to invalidated or updated with the current value to maintain consistency of shared data block.
- DSM must keep track of location of all copies of data blocks in this.



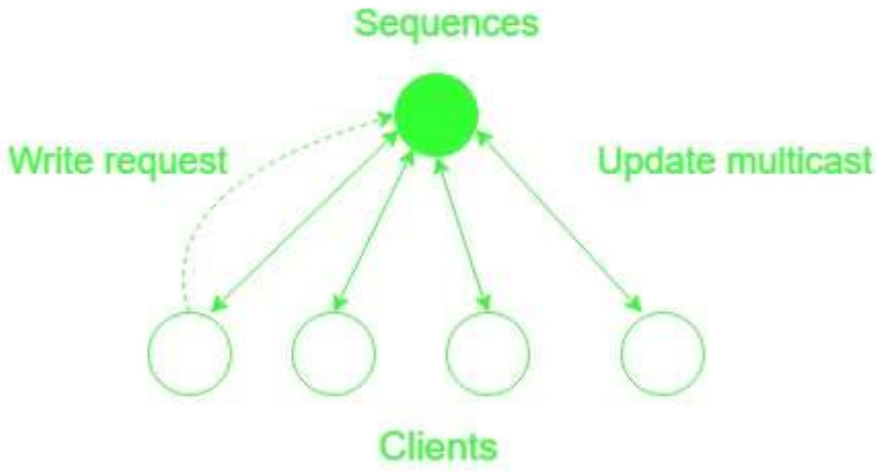
Write operation in Read Replication algorithm

Source: www.geeksforgeeks.org

8.2 Implementing Distributed Shared Memory

Full Replication Algorithm:

- It is an extension of read replication algorithm which allows multiple nodes to have both read and write access to shared data blocks.
- Since many nodes can write shared data concurrently, the access to shared data must be controlled to maintain it's consistency.
- To maintain consistency, it can use a gap free sequences in which all nodes wishing to modify shared data will send the modification to sequencer which will then assign a sequence number and multicast the modification with sequence number to all nodes that have a copy of shared data item.



Write operation in Full Replication algorithm

Source: www.geeksforgeeks.org

8 Distributed Shared Memory Systems and Programming

[Weightage(15%): Approx. 10-11 Marks out of 70 Marks]

- Explain Distributed shared memory
- Explain the advantages/benefits of Distributed shared memory
- Explain Central Server Algorithm to implement Distributed shared memory
- Explain Migration Algorithm to implement Distributed shared memory
- Explain Read Replication Algorithm to implement Distributed shared memory
- Explain Full Replication Algorithm to implement Distributed shared memory