

# Process Management

# Process

---

- *Communication* takes place between *processes*.
- **But, what's a process?** “*A program in execution*”.
- **Traditional operating systems:** concerned with the “local” management and scheduling of processes.
- **Modern distributed systems:** a number of other issues are of equal importance.
- There are three main areas of study:
  1. Threads and virtualization within clients/servers
  2. Process and code migration
  3. Software agents

# Process management

---

- Conventional(Centralied) OS:
  - deals with the mechanisms and policies for sharing the processor of the system among all processes
- Distributed operating system:
  - To make best possible use of the processing resources of the entire system by sharing them among all processes

# Process management cont...

---

- Three concepts to achieve this goal:
  - Processor allocation
    - Deals with the process of deciding which process should be assigned to which processor
  - Process migration
    - Deals with the movement of a process from its current location to the processor to which it has been assigned
  - Threads
    - Deals with fine-grained parallelism for better utilization of the processing capability of the system

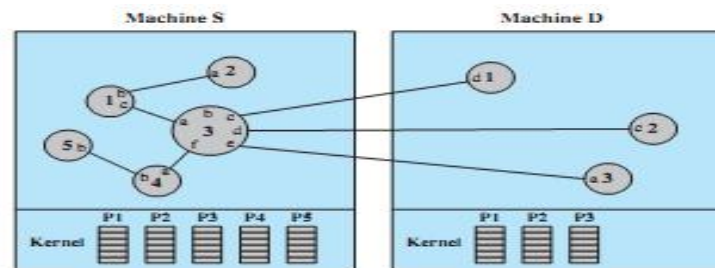
# Process Migration

---

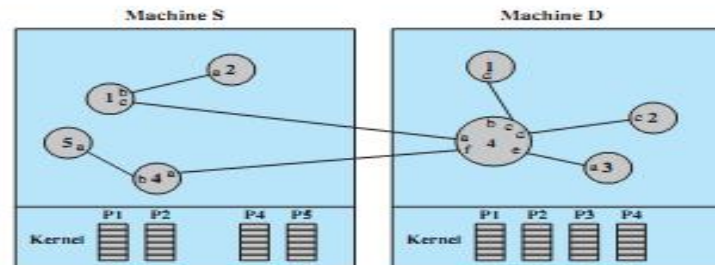
- Under certain circumstances, in addition to the usual passing of data, *passing code* (even while it is executing) can greatly simplify the design of a DS.
- Code migration can be inefficient and very costly.
- So, why migrate code? biggest single reason: **better performance**.
- The idea is to move a compute-intensive task from a *heavily loaded* machine to a *lightly loaded* machine “on demand” and “as required”.

# Process Migration

- Process Migration:
  - The act of transferring a process between two machines during its execution
  - Relocation of a process from its current location (the source node) to another node (the destination node)



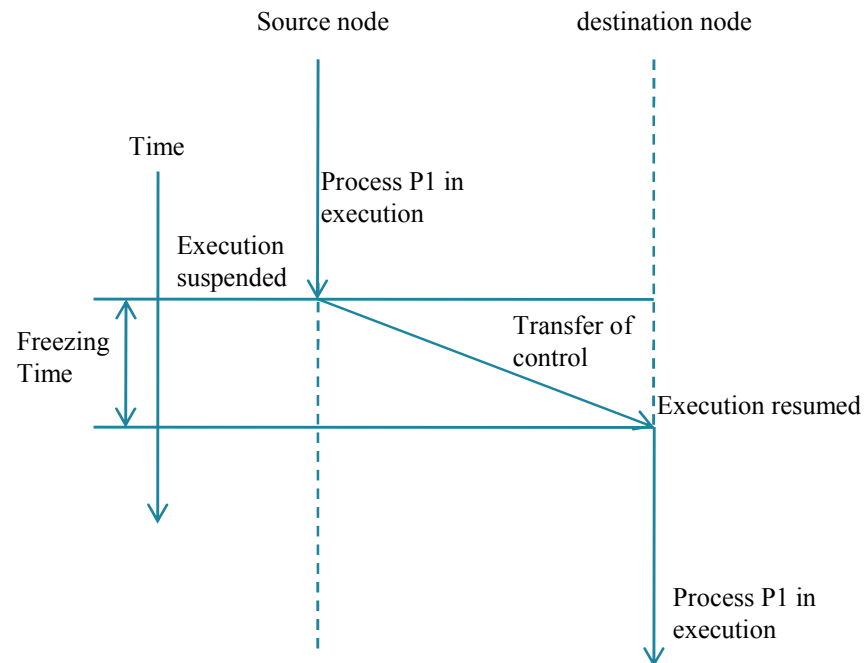
(a) Before migration



(b) After migration

# Process Migration cont...

- The flow of execution of a migrating process:



# Process Migration cont...

---

- It involves following major steps: same as Virtual Machine migration
  1. Selection of a process and node that should be migrated
  2. Selection of the destination node to which the selected process should be migrated
  3. Actual transfer of the selected process to the destination node
- Step 1,2 are taken care by process migration policy
- Step 3 is taken care by process migration mechanism
- Two types:
  - Preemptive process migration
    - Process may be migrated during the course of its execution
  - Non preemptive process migration
    - Process may be migrated before it starts executing on its source node



# Desirable features of a good process migration mechanism

- Transparency
  - Object access level
    - is minimum requirement of system to support non-preemptive process
    - access to objects such as files, devices can be done in a location-independent manner
    - allow free initiation of a programs at an arbitrary node
  - System call and interprocess communication level
    - Hence, migrated process does not continue to depend upon its source node
    - All system calls, inter process communication are location independent
    - support preemptive process migration facility

# Desirable features cont...

---

- Minimal interference
  - Should cause minimal interference to the progress of process and system
  - Can be done by minimizing freezing time
  - Freezing time: a time period for which the execution of the process is stopped for transferring its information to the destination node

# Desirable features cont...

---

- Minimal residual dependencies
  - Migrated process should not continue to depend on its previous node once it has started executing on new node
  - Otherwise problems will occur:
    - Migrated process continue to impose a load on its previous node
    - A failure or reboot of the previous node will cause the process to fail
- Efficiency
  - Time required of migrating a process
  - The cost of locating an object
  - The cost of supporting remote execution once the process is migrated

# Desirable features cont...

---

- Robustness
  - The failure of a node other than the one on which a process is currently running should not affect the execution of that process
- Communication between Coprocesses of a job
  - Parallel processing among the processes of a single job
  - Processes can distributed over several nodes
  - These coprocesses be able to directly communicate irrespective of their location

# Process migration mechanisms

---

- Process migration involves proper handling of several sub-activities to meet the good process migration mechanism requirements.
- Four major sub-activities
  - Freezing and restarting the process
  - Transfer of process's address space
  - Forwarding messages meant for the migrant process
  - Handling communication between cooperating processes
- There are process migration mechanisms for handling each of these sub-activities

# Process migration mechanisms

- Mechanisms for freezing and restarting a process
  - for preemptive process: take a “snapshot of the process state”
  - 1<sup>st</sup> step: freeing the process
    - Execution of the process is suspended and all external interactions are postponed
  - 2<sup>nd</sup> step: state information is transferred to its destination node
  - 3<sup>rd</sup> step: process is restarted using state information on destination node
- Immediate and Delayed blocking of the process
  - Before process can be frozen, its execution must be blocked
  - may be blocked immediately or delayed
  - if the process is not executing a system call --- can be immediately blocked
  - if the process is executing a system call but is sleeping at an interruptible priority waiting for a kernel event to occur, it can be immediately blocked from further execution
  - if the process is executing a system call but is sleeping at non-interruptible priority waiting for a kernel event to occur, it can not be blocked, has to be delayed until the system call is complete.

# Mechanisms for freezing and restarting a process

- Fast and Slow I/O operation
  - frozen after the completion of all fast I/O operations
  - What about slow I/O operations???
- Information about the open files
  - No problem for DS with network transparent execution environment
  - What about UNIX like systems??? Identify files by full pathnames
    - creation of link – link is created to the file and pathname of link is used as an access point
    - Reconstruction of file's path when required – modification of kernel
  - What about frequently used files like commands???
  - What about temporary files?

# Mechanisms for freezing and restarting a process

---

- Reinstating the process on its destination node
  - Creation of a new process – empty process state
  - Process identifier
  - Once all the state information of the migrating process has been transferred and copied into the empty process state, the new copy of the process is unfrozen and the old copy is deleted.
  - Thus the process is restarted on its destination node in whatever state it was in before migrated.



# Address Space Transfer Mechanisms

- Address Space Transfer Mechanisms
  - Information to be transferred from source node to destination node:
    - Process's state information
      - Consists contents of register, scheduling information, memory tables, I/O buffers, interrupt signals, I/O states, process identifier, information about I/O files, process's user and group identifiers
    - Process's address space
      - Consists code, data, stack of the program
  - Difference between the size of process's state information (few kilobytes) and address space (several megabytes)
  - Time taken to transfer address space is more
  - Possible to transfer the address space without stopping its execution
  - Not possible to resume execution until the state information is fully transferred

# Address Space Transfer Mechanisms

---

- Three methods for address space transfer
  - Total Freezing/postcopy
  - Pretransferring/live migration /precopy
  - Transfer on reference/on-demand

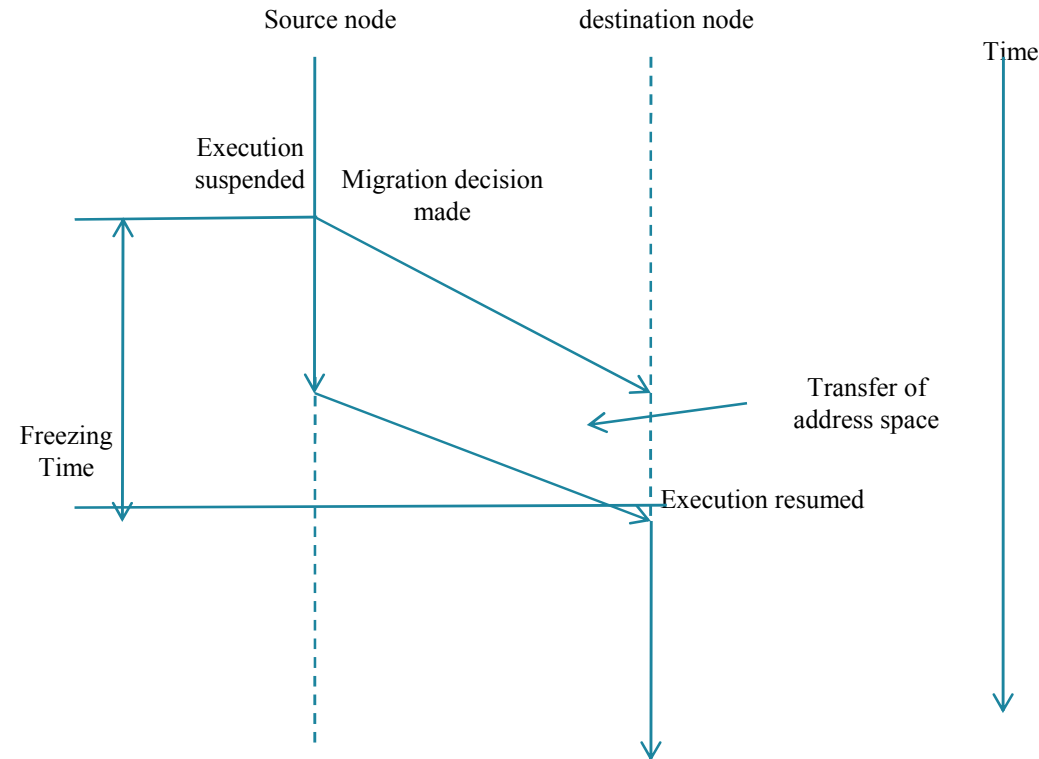
# Address Space Transfer Mechanisms

---

- Total Freezing:
  - Process execution is stopped while its address space is being transferred
  - Simple and easy to implement
  - Disadvantages:
    - Process is suspended for a long time during migration
    - Not suitable for interactive process, the delay can be occurred

# Address Space Transfer Mechanisms

- Total Freezing:



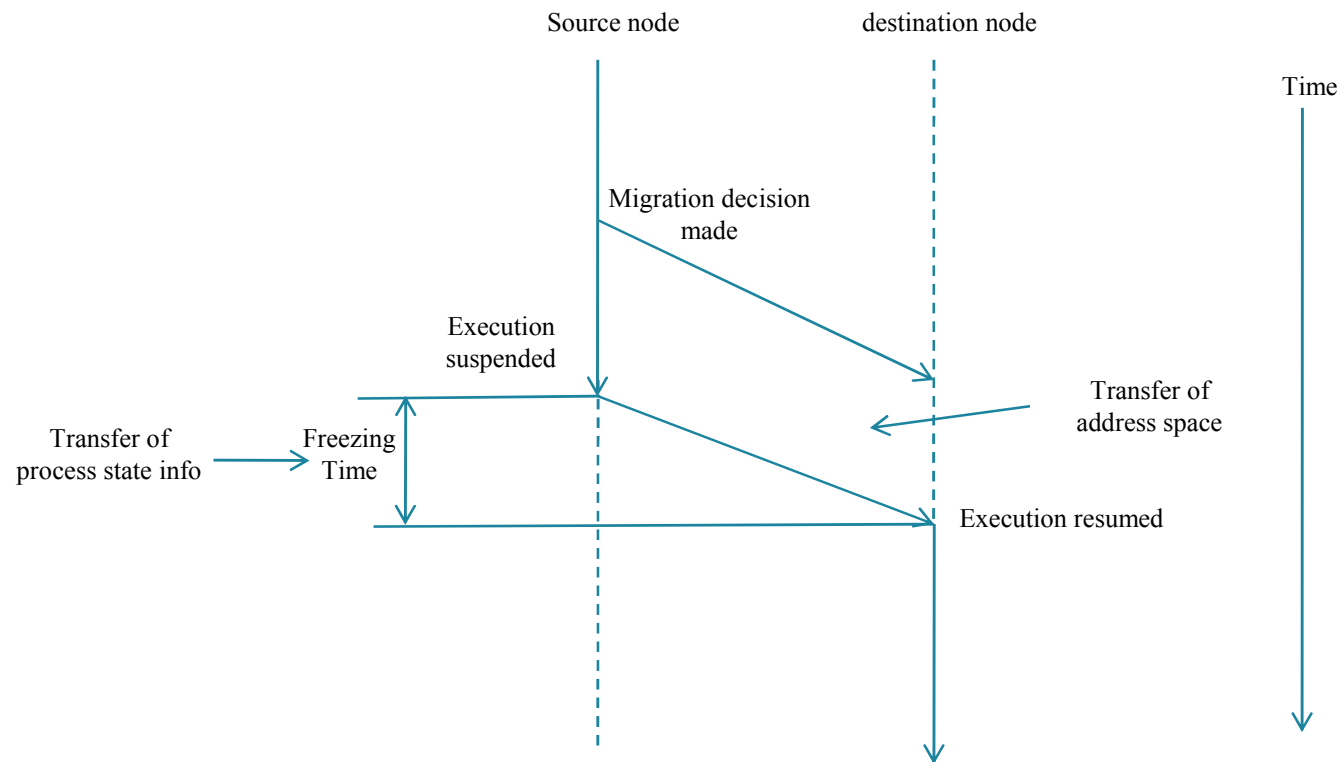
# Address Space Transfer Mechanisms

---

- Pretransferring (precopying):
  - Address space is transferred while the process is still running on the source node
  - Initial transfer of the complete address space followed by repeated transfers of the pages modified during previous transfer
  - Remaining modified pages are retransferred after the **process is frozen for transferring its state information**
  - Pretransfer operation is executed at a higher priority than all other programs on the source node

# Address Space Transfer Mechanisms

- Pretransferring (precopying):



# Address Space Transfer Mechanisms

---

- Advantage:
  - freezing time is reduced
- Disadvantage:
  - Total time of migration is increased due to the possibility of redundant page transfers
  - Redundant pages are pages that are transferred more than once during pretransferring because they become dirty while the pretransfer operation is being performed.

# Address Space Transfer Mechanisms

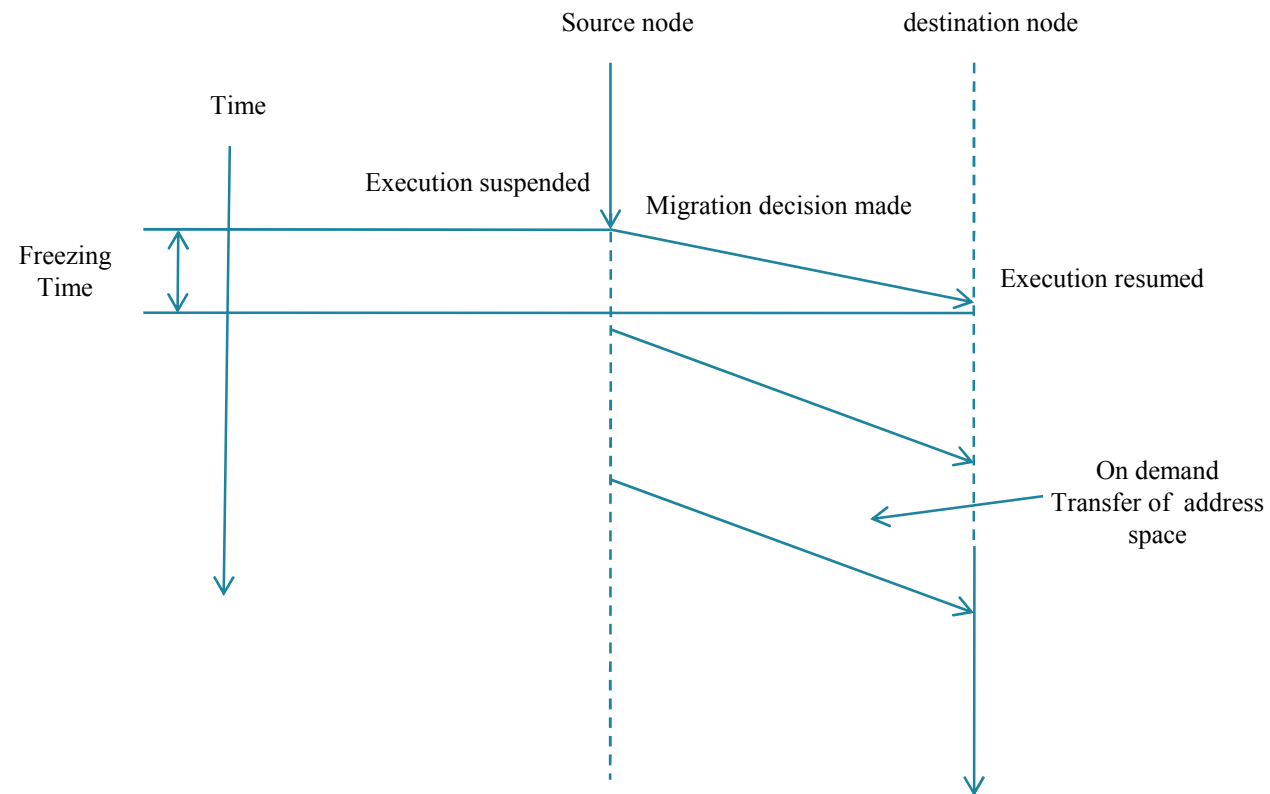
---

- Transfer on reference
  - Based on the assumption that the process tends to use only a relatively small part of their address space while executing.
  - A page of the address space is transferred from its source node to destination node only when referenced
  - Demand-driven copy-on-reference approach
  - Switching time is very short and independent of the size of the address space



# Address Space Transfer Mechanisms

- Transfer on reference



# Address Space Transfer Mechanisms

---

- Disadvantages:
  - Not efficient in terms of cost
  - Imposes continued load on process's source node
  - Results in failure if source node fails or reboots

# Message forwarding mechanisms

---

- In moving a process, it must be ensured that all pending, en-route and future messages arrive at the process's new location.
- Classification of the messages to be forwarded to the migrant process new location:
  - Type 1: Messages received at the source node after the process's execution has been stopped on its source node and process's execution has not yet been started on its destination node
  - Type 2: Message received at the source node after the process's execution has started on its destination node
  - Type 3: Messages that are to be sent to the migrant process from any other node after it has started executing on the destination node

# Message forwarding mechanisms

## 1. Mechanism of Resending the Message

- Used in V-system and Amoeba.

Message Type	V-System	Amoeba
Type 1 or Type 2	<ul style="list-style-type: none"><li>• Simply dropped the message and sender is prompted to resend it to the process's new node</li><li>• Sender will retry until successful receipt of a reply</li></ul>	Source node's kernel sends the message to sender Type 1: "try again later, this process is frozen" Type 2: "this process is unknown at this node"
Type 3	Sender does a "broadcast mechanism" to find the new location of the process	

- Drawback
  - It is nontransparent to the processes interacting with the migrant process

# Message forwarding mechanisms

## 2. Origin Site Mechanism

- Used in AIX's TCF (Transparent Computing Facility) and Sprite
- The process identifier of these systems has the process's origin site (or home node) embedded in it
- Each site is responsible for keeping information about the current location of all the processes created on it
- Therefore, a process's current location can be simply obtained by consulting its origin site
- Messages are sent to the origin site first and from there they are forwarded to the current location
- Drawbacks:
  - not good from reliability point of view
    - Failure of origin site will disrupt the message-forwarding mechanisms
  - continuous load on migrant process's original site

# Message forwarding mechanisms

## 3. Link-Traversal Mechanism

- Used in DEMOS/MP
- Uses message queue for storing messages of type 1- redirect the messages
  - A message queue for the migrant process is created on its source node.
  - Once process is established on the destination node, all messages in the queue are sent to the destination node as a part of the migration procedure
- Use of link (a forwarding address) for messages of type 2 and 3 to redirect the messages
  - Link is left at the source node pointing to the destination node
- Link has two components:
  - Process identifier: id of the node+ local id for process ->never changes
  - last known location of the process -> changes
- Migrated process is located by traversing a series of links

# Message forwarding mechanisms

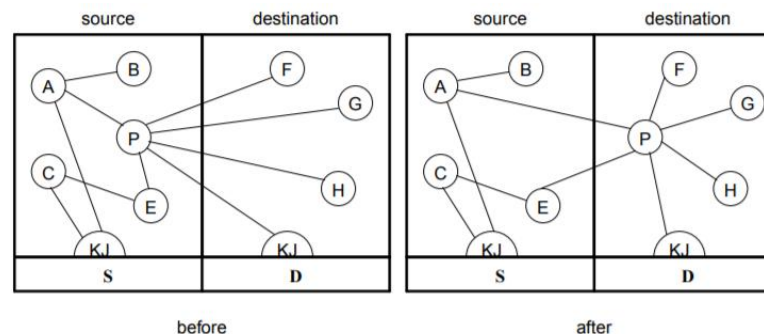
---

- Link-Traversal Mechanism
- Drawbacks:
  - poor efficiency
    - Several links may have to be traversed to locate a process from a node
  - poor reliability
    - If any node in the chain of links fails, the process cannot be located

# Message forwarding mechanisms

## 4. Link Update Mechanism

- Used in Charlotte
- Processes communicate via location independent links
- Messages of type 1 and 2 are forwarded to the destination node by source node
- During the transfer phase, the source node sends link update message to all relevant kernels controlling all of the migrant process's communication partners
- Link update message tell the new address of the migrant process -> in parallel so not expensive
- Type 3 messages are sent directly to the process's destination node





# Mechanisms for handling co-processes

---

- Important issue is the necessity to provide efficient communication between a process (parent) and its sub-processes (children)
- Two different mechanisms
  - Disallowing separation of Co-processes
  - home node or origin site concept

# Mechanisms for handling co-processes

- Disallowing separation of Co-processes: some of UNIX based system
  - By disallowing the migration of processes that wait for one or more of their children to complete.
  - By ensuring that when a parent process migrates, its children processes will be migrated along with it
    - Concept of logical host
      - Address spaces and their associated processes are grouped into logical host
      - Migration of a process is actually migration of logical host
  - Drawback:
    - Does not allow parallelism within jobs, which is achieved by assigning various tasks of a job to the different nodes of the system and execute them simultaneously
    - Overhead is large when logical host contains several processes

# Mechanisms for handling co-processes

---

- Home node or origin site concept: V system
  - Complete freedom of migrating a process or its sub-processes independently and executing them on different nodes
  - All communications between a parent process and children processes take place via home node
- Drawback:
  - The message traffic and the communication cost increase
  - Lost communication of origin node is fails

# Advantages of process migration

---

- Reducing average response time of processes
  - avg. response time is increase as load increase on to the node
  - Process is migrated from heavily loaded node to a idle/underutilized node
- Speeding up individual jobs
  - Execute tasks of a job concurrently
  - To migrate a job to a node having faster CPU
- Gaining higher throughput
  - In a system that does not support process migration, CPUs of all nodes are not fully utilized.
  - In a system that support process migration, CPUs of all nodes can be better utilized by using suitable load balancing policy

# Advantages of process migration

---

- Utilizing resources effectively
  - Capabilities of the various resources such as CPU, printers, storage devices etc.
  - Depending on the nature of the process, it can be migrated to the most suitable node
- Reducing network traffic
  - Migrate the process closer to the resources it is using most heavily
  - To migrate and cluster two or more processes which frequently communicate with each other, on the same node
- Improving system reliability
  - Migrating critical process to more reliable node
- Improving system security
  - A sensitive process may be migrated and run on the secure node that is not directly accessible to general users thus improve the security of that process

# Assignment – Paper writing..

---

- 1-Live VM migration for memory intensive applications
- 2-Live VM migration for compute/CPU intensive applications
- 3-Precopy live VM migration
- 4-Postcopy live VM migration
- 5-Hybrid(Pre+Post) live VM migration
- 6-Secure live VM migration
- 7-Live migration of containerized applications OR Live migration of docker containers
- 8- Energy-aware VM migration

Papers – from good journals/conference proceedings of  
IEEE, ACM, Springer, Wiley, - [scholar.google.com](https://scholar.google.com)