CSE 4733/6733 - Operating System 1

Stephen A. Torri, Ph.D.

Mississippi State University

CSE 4733/6733 - Operating System 1

Stephen A. Torri, Ph.D.

Mississippi State University

Paging[1]

Definition

Paging is a memory management scheme that eliminates the need for a contiguous allocation of physical memory by retrieving processes in the form of pages from the secondary storage into the main memory.

Paging[1]
Definition
Paging is a memory management scheme that eliminates the need for a contiguous allocation of physical memory by retrieving processes in the form of pages from the secondary storage into the main memory.

1. The basic purpose of paging is to separate each procedure into pages. Additionally, frames will be used to split the main memory. This scheme permits the physical address space of a process to be non – contiguous.
2. The paging splits the logical space into fixed-sized blocks.

Paging[1]
Features

Features of paging:

- Mapping logical address to physical address.
- Page size is equal to frame size.
- Number of entries in a page table is equal to the number of pages in logical address space.
- The page table entry contains the frame number.
- All the entries in the page table of the processes are placed in the main memory.

Paging[1]
Features
Features of paging:
- Mapping logical address to physical address.
- Page size is equal to frame size.
- Number of entries in a page table is equal to the number of pages in logical address space.
- The page table entry contains the frame number.
- All the entries in the page table of the processes are placed in the main memory.

1. Advantages of Paging:
   - Easy-to-use memory management algorithm
   - No need for external Fragmentation
   - Swapping is easy between equal-sized pages and page frames.
2. Disadvantages of Paging:
   - May cause Internal fragmentation
   - Page tables to consume additonal memory.
   - Multi-level paging may lead to memory reference overhead.

Frames[1]

Definition

The Physical Address Space is conceptually divided into several fixed-size blocks called frames.

Frames[1]
Definition
The Physical Address Space is conceptually divided into several fixed-size blocks called frames.

1. The mapping from virtual to physical address is done by the memory management unit (MMU), which is a hardware device, and this mapping is known as the paging technique.

Paging: Logical Address Components[1]

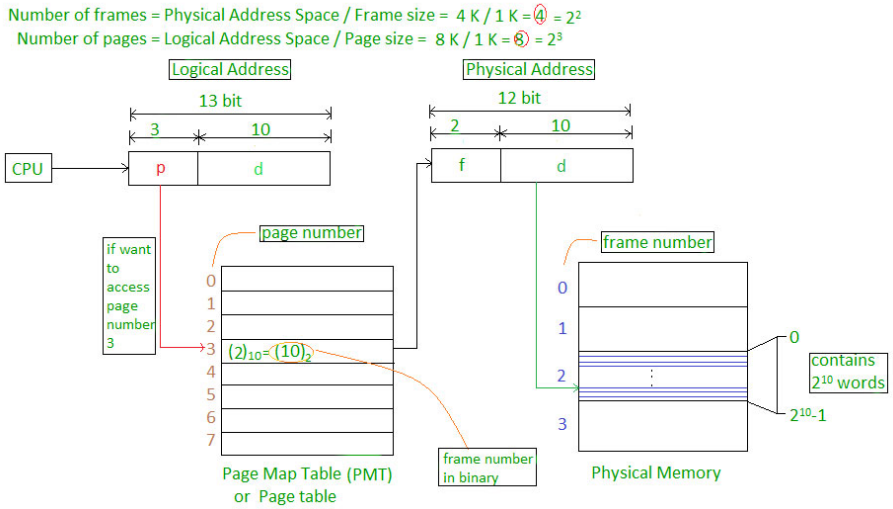
- **Page number(p)**: Page Number is used to specify the specific page of the process from which the CPU wants to read the data. and it is also used as an index to the page table.
- **Page offset(d)**: Page offset is mainly used to specify the specific word on the page that the CPU wants to read.

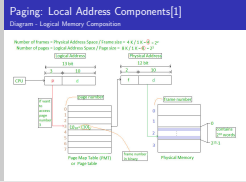
1.

Paging: Logical Address Components[1]
- **Page number(p)**: Page Number is used to specify the specific page of the process from which the CPU wants to read the data. and it is also used as an index to the page table.
- **Page offset(d)**: Page offset is mainly used to specify the specific word on the page that the CPU wants to read.

Paging: Local Address Components[1]

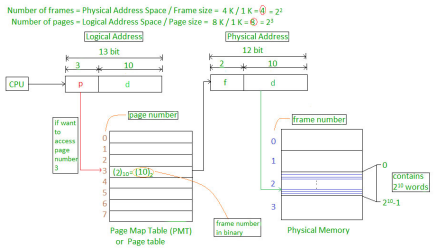
Diagram - Logical Memory Composition





Paging: Page Table[1]

Diagram - Logical Memory Composition



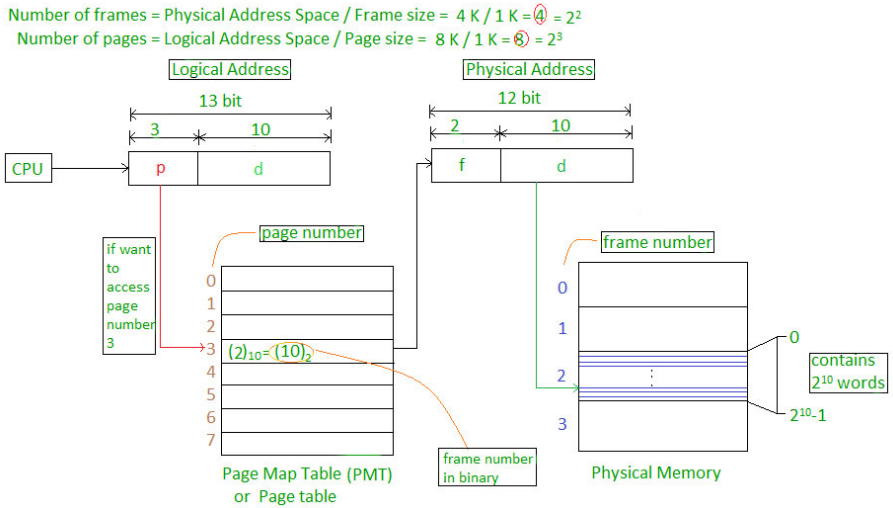
- **Page Table**: Contains the base address of each page in the physical memory.
- **Physical Memory Address**: Composed of the base address combined with the page offset. This value is set to the Memory Management Unit (MMU).

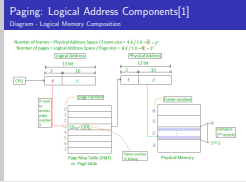
Paging: Page Table[1]
Diagram - Logical Memory Composition
Page Table: Contains the base address of each page in the physical memory.
Physical Memory Address: Composed of the base address combined with the page offset. This value is set to the Memory Management Unit (MMU).

1. Logical address = *page_number* + *page_offset*

Paging: Logical Address Components[1]

Diagram - Logical Memory Composition





Paging: Page Table[1]

Logical Memory Composition

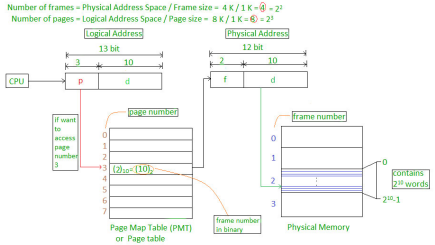
The Page table mainly contains the base address of each page in the Physical memory. To access a location in memory:

1. The base address is then combined with the page offset to define the physical memory address, which is then sent to the memory unit.
2. Thus page table mainly provides the corresponding frame number (base address of the frame) where that page is stored in the main memory.

Paging: Page Table[1]
Logical Memory Composition
The Page table mainly contains the base address of each page in the Physical memory. To access a location in memory:
1. The base address is then combined with the page offset to define the physical memory address, which is then sent to the memory unit.
2. Thus page table mainly provides the corresponding frame number (base address of the frame) where that page is stored in the main memory.

Paging[1]

Diagram - Physical Memory Composition

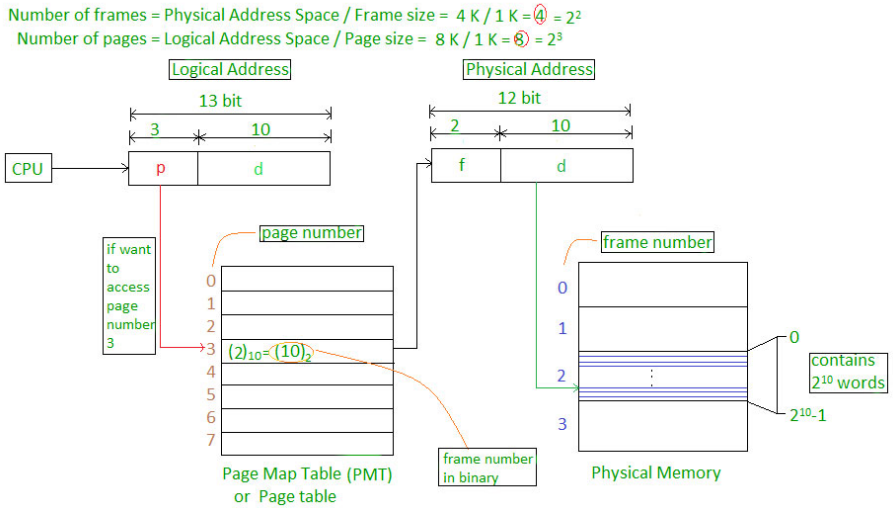


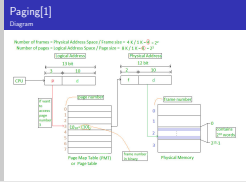
- **Frame number(f)**: Number of bits required to represent the frame of Physical Address Space or Frame number.
- **Frame offset(d)**: Number of bits required to represent a particular word in a frame or frame size of Physical Address Space or word number of a frame or frame offset.

Paging[1]
Diagram - Physical Memory Composition
- **Frame number(f)**: Number of bits required to represent the frame of Physical Address Space or Frame number.
- **Frame offset(d)**: Number of bits required to represent a particular word in a frame or frame size of Physical Address Space or word number of a frame or frame offset.

Paging[1]

Diagram





Paging[1]

Calculating number of pages

The number of pages in physical memory is calculated by:

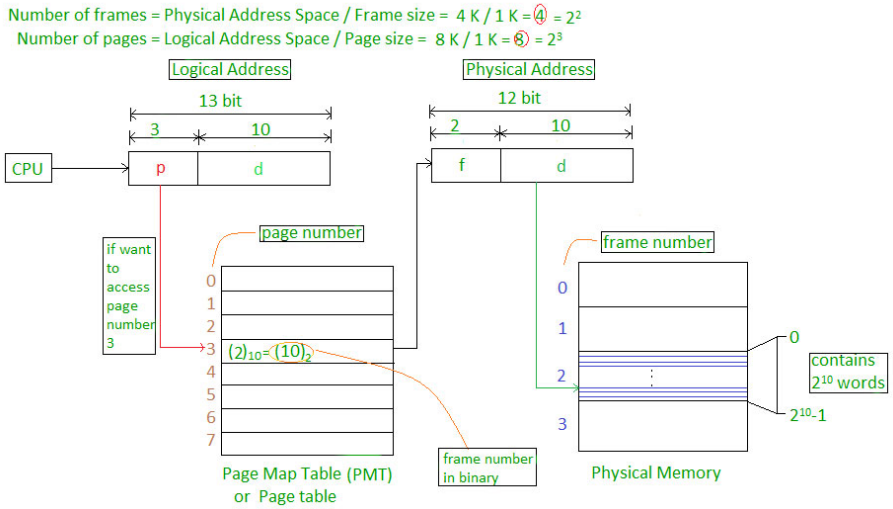
$$\begin{aligned}\text{Number of frames} &= \frac{LogicalAddressSpace}{PageSize} \\ &= \frac{8KB}{1KB} \\ &= 8 \\ &= 2^3\end{aligned}$$

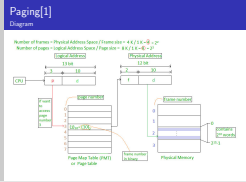
Paging[1]
Calculating number of pages
The number of pages in physical memory is calculated by:
$\text{Number of frames} = \frac{LogicalAddressSpace}{PageSize}$
$= \frac{8KB}{1KB}$
$= 8$
$= 2^3$

1. 3 bits will be used as page index.

Paging[1]

Diagram





Paging[1]
Calculating number of frames

The number of frames in physical memory is calculated by:

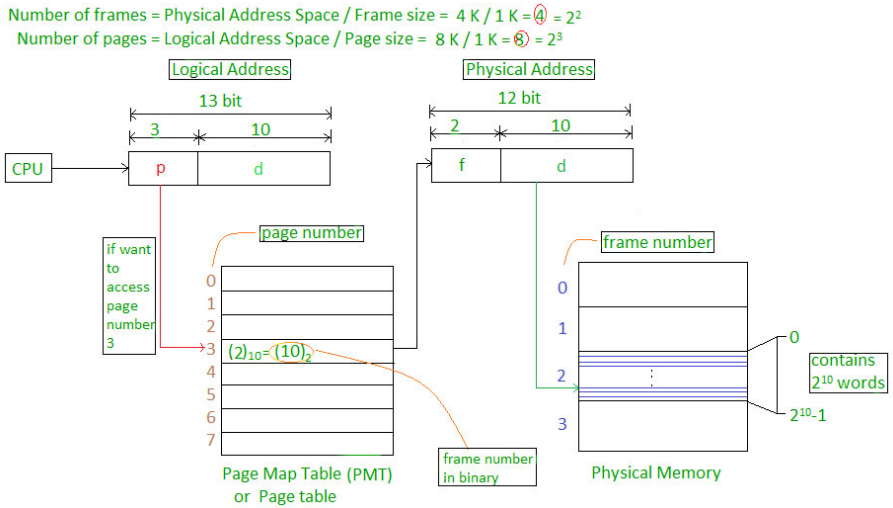
$$\begin{aligned}\text{Number of frames} &= \frac{PhysicalAddressSpace}{FrameSize} \\ &= \frac{4KB}{1KB} \\ &= 4 \\ &= 2^2\end{aligned}$$

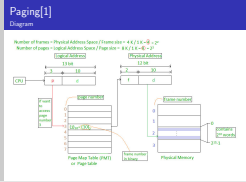
Paging[1]
Calculating number of frames
The number of frames in physical memory is calculated by:
$\text{Number of frames} = \frac{PhysicalAddressSpace}{FrameSize} = \frac{4KB}{1KB} = 4 = 2^2$

1. 2 bits will be used as a frame index

Paging[1]

Diagram





Paging Example[2]

Notes:

- Page table can be implemented as a set of registers in main memory.
- If any process requires n pages, then at least n frames are required.

Steps:

1. Process is loaded into the first frame listed in the free frame list.
2. The frame number is put into the page table.
3. OS gives a copy of the page tabe for each process.
4. This copy is also used by the CPU dispatcher to define the hardware page table whenever a process is to be allocated to the CPU.

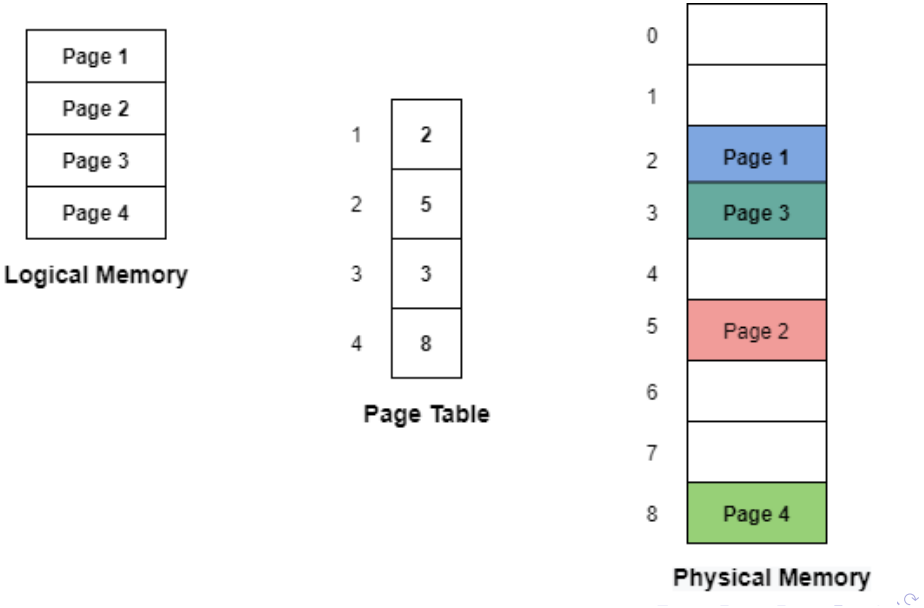
Paging Example[2]

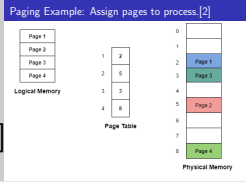
Notes:
- Page table can be implemented as a set of registers in main memory.
- If any process requires n pages, then at least n frames are required.

Steps:
1. Process is loaded into the first frame listed in the free frame list.
2. The frame number is put into the page table.
3. OS gives a copy of the page tabe for each process.
4. This copy is also used by the CPU dispatcher to define the hardware page table whenever a process is to be allocated to the CPU.

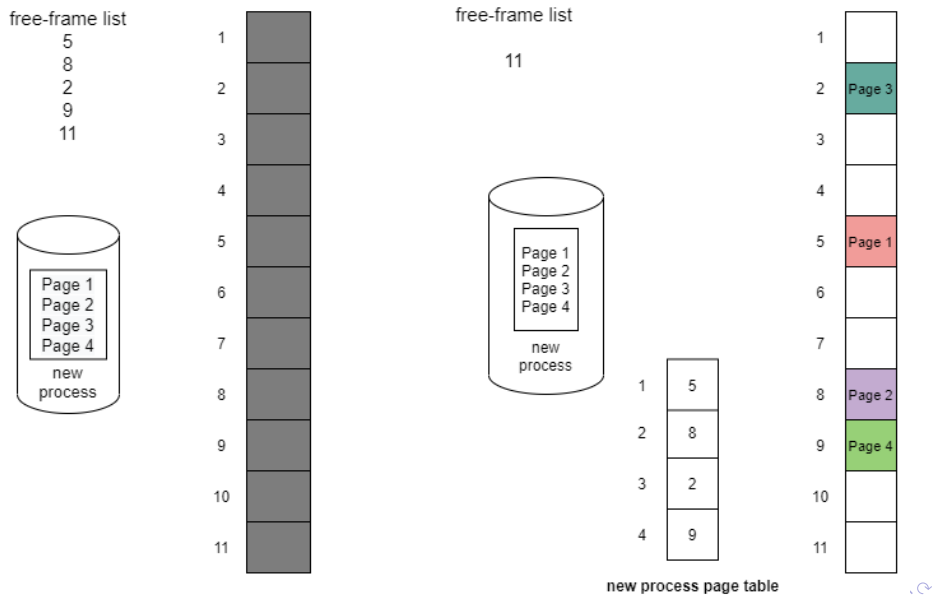
1. As the size of registers is limited, the page table size is usually large and, therefore, can be kept in the main memory.
2. There is no External fragmentation caused due to this scheme; Any free frame can be allocated to any process that needs it. But the internal fragmentation is still there inside each frame.
3. The frame table is a data structure that keeps the information of which frames are allocated or which frames are available and many more things. This table mainly has one entry for each physical page frame.
4. The Operating system maintains a copy of the page table for each process in the same way as it maintains a copy of the instruction counter and registers contents. Also, this copy is used to translate logical addresses to physical addresses

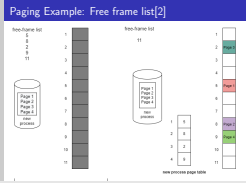
Paging Example: Assign pages to process.[2]





Paging Example: Free frame list[2]





References I


Unknown.
Paging in operating system, 2023.


Unknown.
Paging in operating systems, 2023.

References I

Unknown.
Paging in operating system, 2023.
Unknown.
Paging in operating systems, 2023.