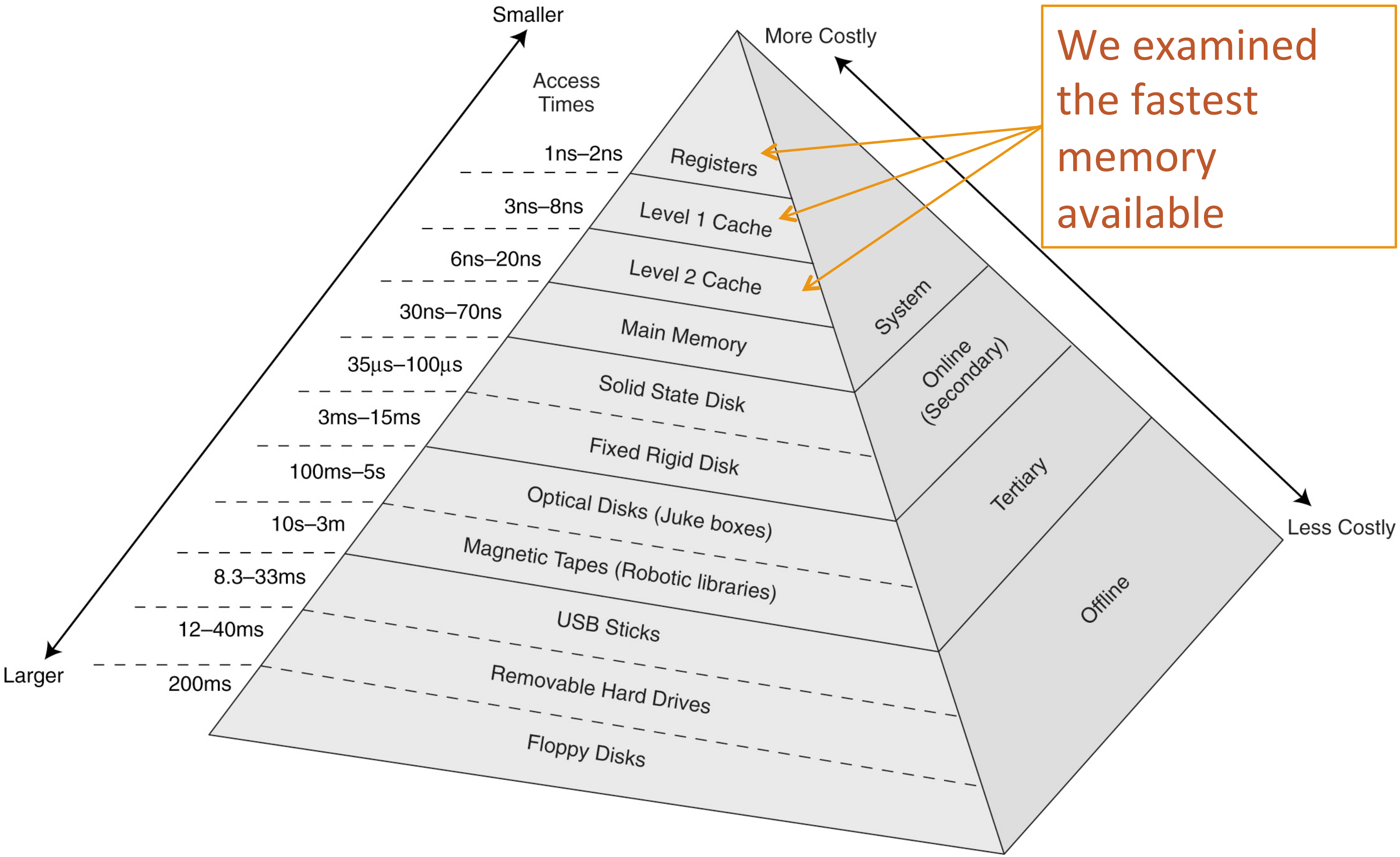
LECTURE 10: VIRTUAL MEMORY AND BASIC BASH SCRIPTING FOR LAB 7

Computer Systems and Networks

Dr. Pallipuram
(vpallipuramkrishnamani@pacific.edu)

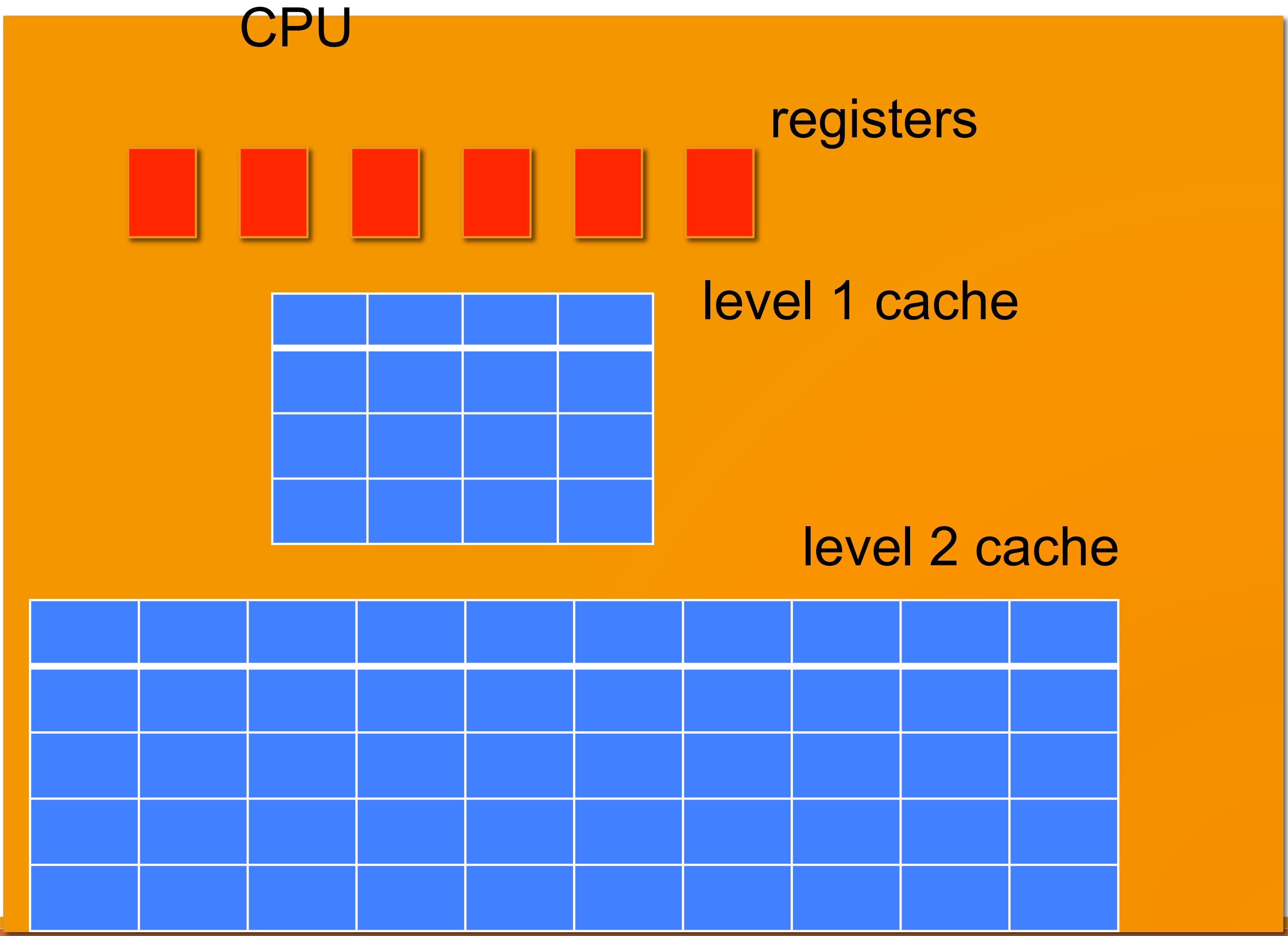
Today's Class

- Virtual Memory
- Basic Bash script automation



We examined the fastest memory available

Multi-level cache hierarchy



CPU

registers

level 1 cache

level 2 cache

Caches are Ubiquitous

L0: Registers act as cache for L1. (Speed: 1 clock cycle) ✓

L1: L1 acts as a cache for L2. (Speed: 5-10 clock cycles) ✓

L2: L2 acts as a cache for L3. (Speed: 20-30 clock cycles) ✓

L3: L3 acts as a cache for main memory. (Speed: 60-80 clock cycles) ✓

main memory: acts as a cache for the disk. (Speed: 100 – 200 clock cycles)

disk: acts as a cache for remote servers. (Speed: lot of cycles!)

Virtual Memory

Virtual Memory is a way that gives your program an illusion of LARGE amount of memory

The main memory (DRAM) acts as the cache for the disk

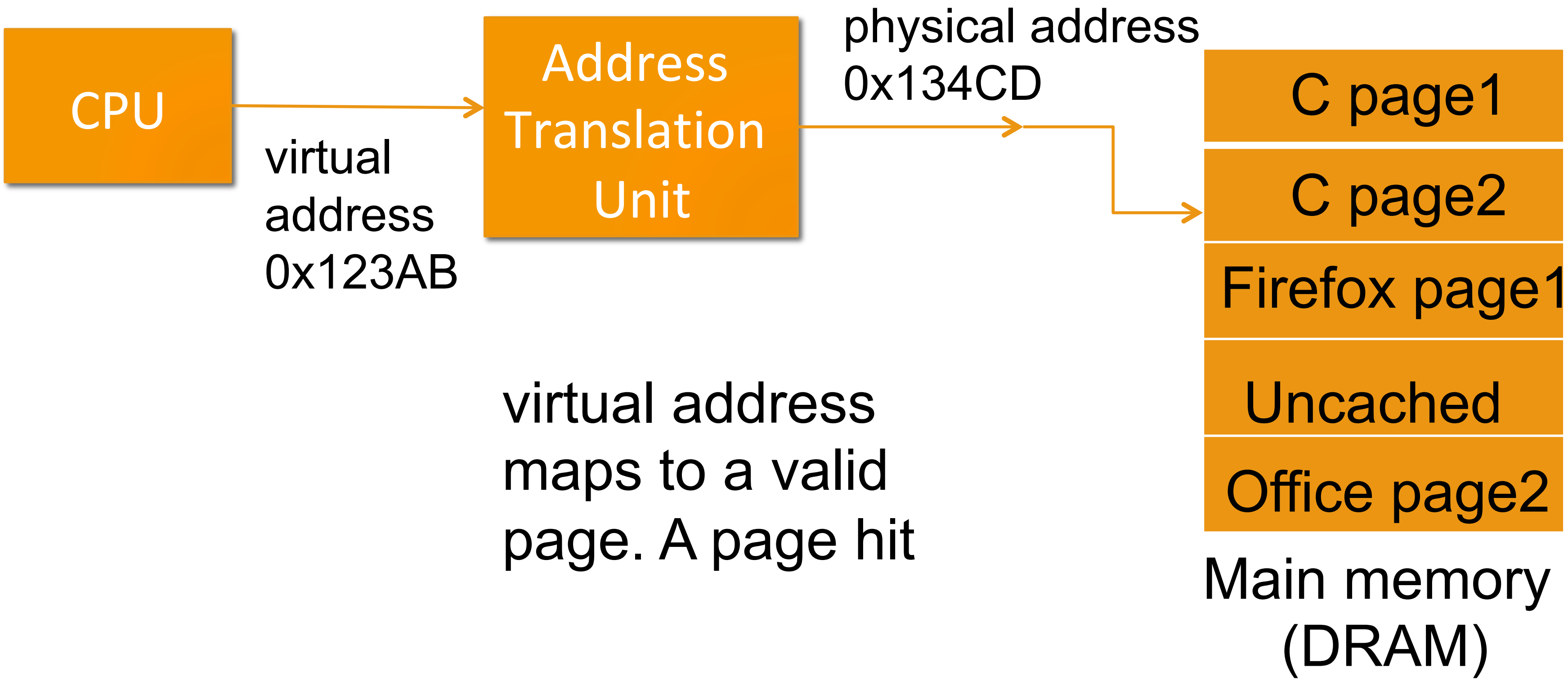
Using Virtual memory, you can have large number of programs active

OS divides your active programs into a number of pages.

The frequently used ones are in DRAM (cached) for faster accesses

Less frequently used ones are on the disk and are brought into the main memory when required

Assume that your C program is executing, and LibreOffice and Firefox are active



CPU

virtual address 0x123AB

Address Translation Unit

physical address 0x134CD

C page1

C page2

Firefox page1

Uncached

Office page2

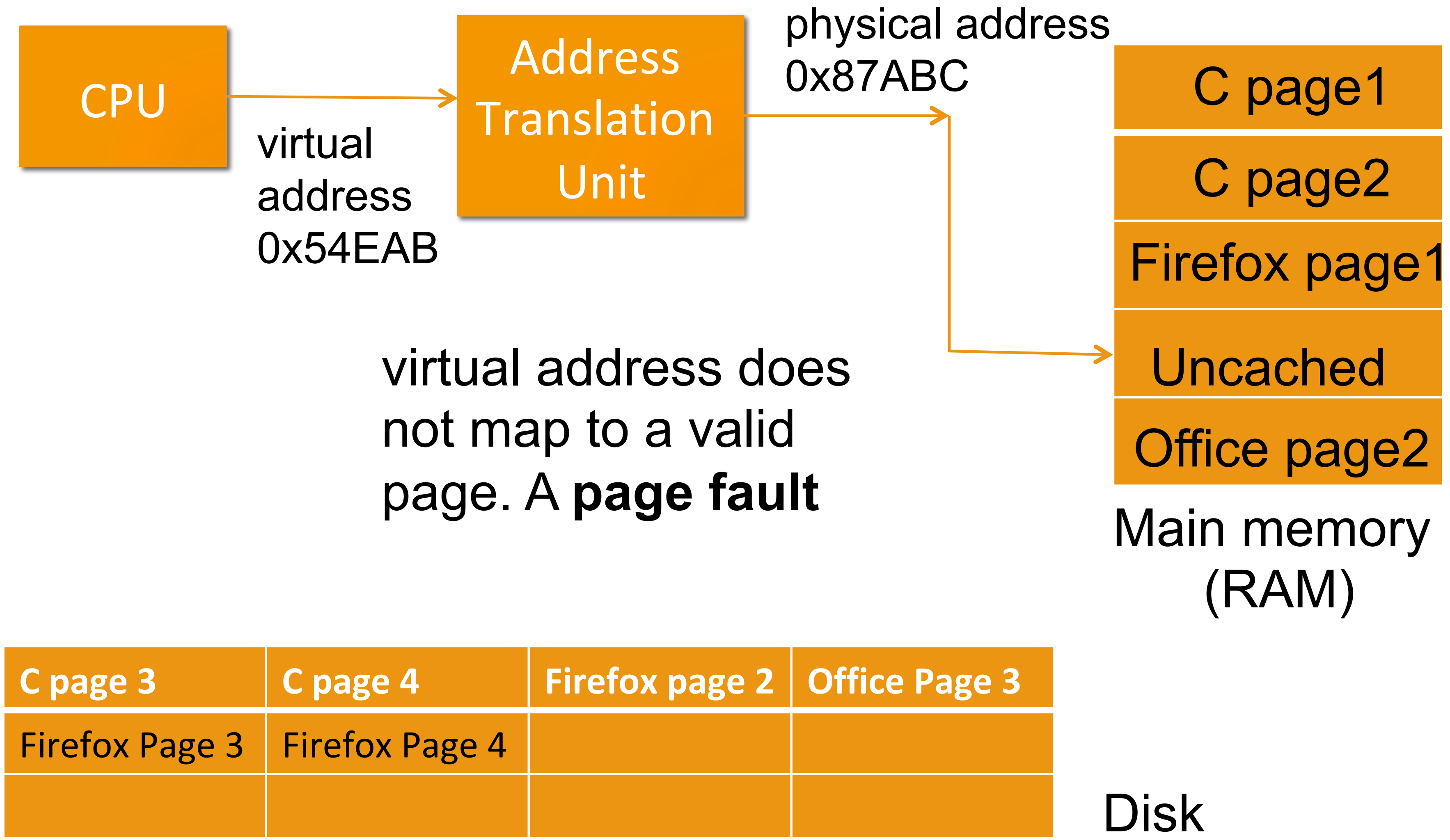
Main memory (DRAM)

virtual address maps to a valid page. A page hit

C page 3	**C page 4**	**Firefox page 2**	**Office Page 3**
Firefox Page 3	Firefox Page 4		

Disk

Assume that your C program is executing and LibreOffice and Firefox are active



CPU

virtual address 0x54EAB

Address Translation Unit

physical address 0x87ABC

C page1
C page2
Firefox page1
Uncached
Office page2

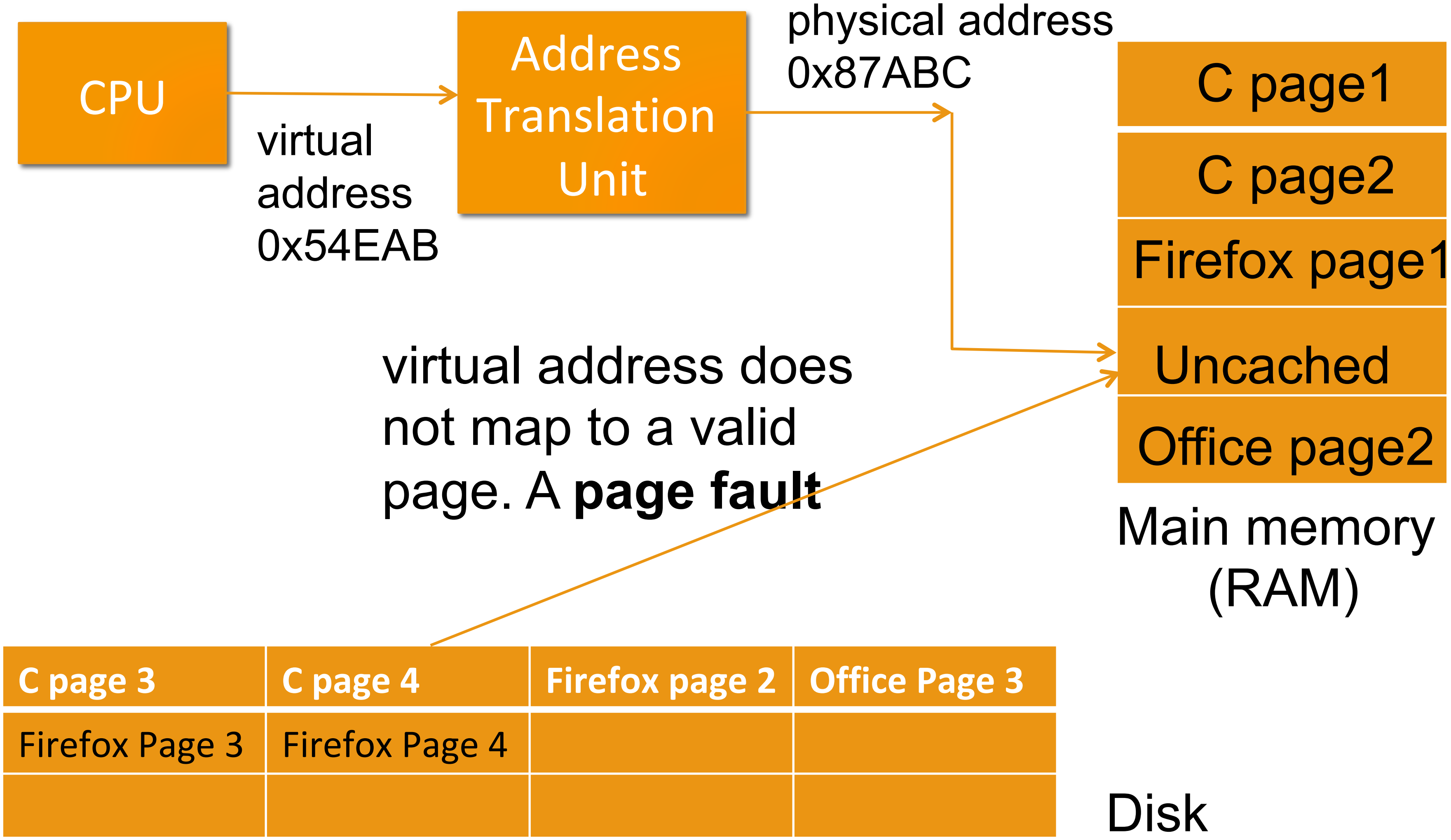
Main memory (RAM)

virtual address does not map to a valid page. A **page fault**

C page 3	**C page 4**	**Firefox page 2**	**Office Page 3**
Firefox Page 3	Firefox Page 4		

Disk

Assume that your C program is executing and LibreOffice and Firefox are active



CPU

virtual address 0x54EAB

Address Translation Unit

physical address 0x87ABC

C page1

C page2

Firefox page1

Uncached

Office page2

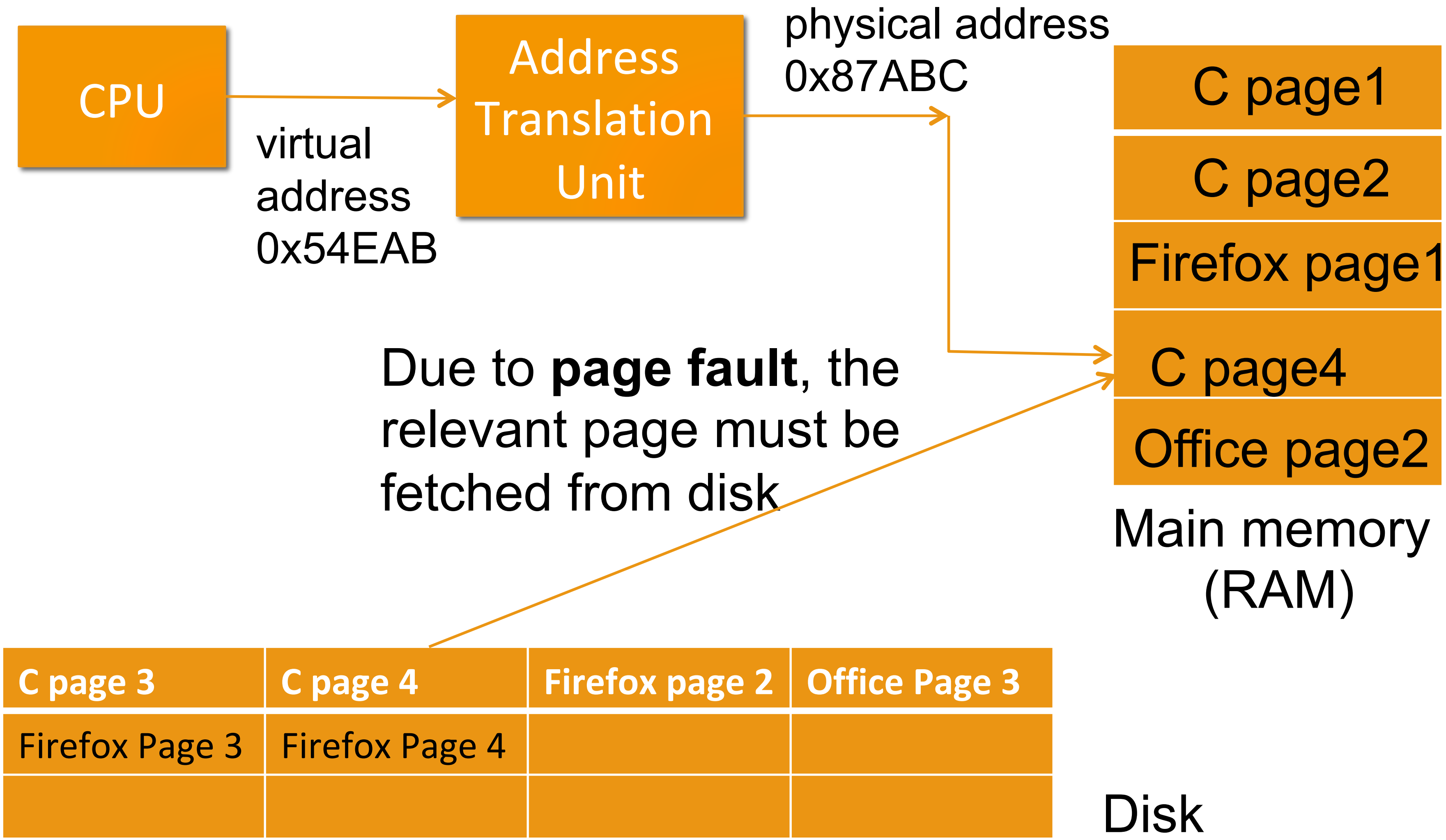
Main memory (RAM)

virtual address does not map to a valid page. A **page fault**

C page 3	**C page 4**	**Firefox page 2**	**Office Page 3**
Firefox Page 3	Firefox Page 4		

Disk

Assume that your C program is executing and LibreOffice and Firefox are active



CPU

virtual address 0x54EAB

Address Translation Unit

physical address 0x87ABC

C page1

C page2

Firefox page1

C page4

Office page2

Main memory (RAM)

Due to **page fault**, the relevant page must be fetched from disk

C page 3	**C page 4**	**Firefox page 2**	**Office Page 3**
Firefox Page 3	Firefox Page 4		

Disk

Think…

Virtual memory space allows you to run several applications concurrently

Does that mean you can keep facebook, spotify, office, and Skype active AND expect your C program to run fast? Why or why not?

Bash Script Automation

FOR LAB 7. IF TIME PERMITS, WE’LL SEE SOME ADVANCED CONCEPTS

Exercise 1 -- Bash Scripting

Every bash script usually begins with a **Shebang (#!)** – It is used to specify the absolute path of the bash

Example: `#!/bin/bash`

Create a folder in your home folder called BASH. cd to BASH.

Gedit a file: mybash1.sh. Add the above shebang and save. Change the mode of `mybash1.sh` to an executable. Recall our Linux exercise

Exercise 2: For Loops in Bash – Version 1

Add this code in `mybash1.sh`

```
#!/bin/bash
for i in 1 2 3 4 5
do
        echo "Welcome number: $i"
done
```

execute: ./mybash1.sh

Exercise 3 – A better for loop

```
#!/bin/bash
for ((i=0;i<12;i++))
do
        echo "Welcome number: $i"
done
```

$ replaces the variable with its value

Create a new file called mybash2.sh. Can you modify the above code to create folders: lab2 to lab12?

Exercise 4 -- Conditionals in bash

```
if ((<some C-like conditional>))
then
<commands>
fi
```

Create a new file called mybash3.sh. Can you modify the above code to create folders: lab02, lab03,..,lab09,.. lab12?

Concept of Arrays in Bash

```
Syntax:
declare -a arrayname=(element1 element2
element3)
```

```
declare -a Unix=('Debian' 'Red hat' 'Red hat'
'Suse' 'Fedora');
```

length of an array: `${#ArrayName[@]}`

example of accessing an element at ith position:

```
${ArrayName[i]}
```

Putting it all together

I want to run the amplification program on Lenna_org_1024.pgm, gaussian width = 11, and for sigma values: 0.3, 0.4, …1.1 (totaling 9 executions). Automate these lines:

```
./amplify Lenna_org_1024.pgm 11 0.3 2
./amplify Lenna_org_1024.pgm 11 0.4 2
./amplify Lenna_org_1024.pgm 11 0.5 2
:
./amplify Lenna_org_1024.pgm 11 1.1 2
```

Exercise 5

I want to run the amplification program on Lenna_org_1024.pgm for sigma values: 0.3, 0.4, …1.1 AND

gaussian widths: 5 7 11 15 19