LLMs as Operating Systems

DS 5110: Big Data Systems

Spring 2025

Lecture 11

Yue Cheng

Some material taken/derived from:
- Intro to LLMs, Andrej Karpathy.

Learning objectives

- Learn how LLM agent systems resemble a modern OS
- Understand how MemGPT's extended, editable context works

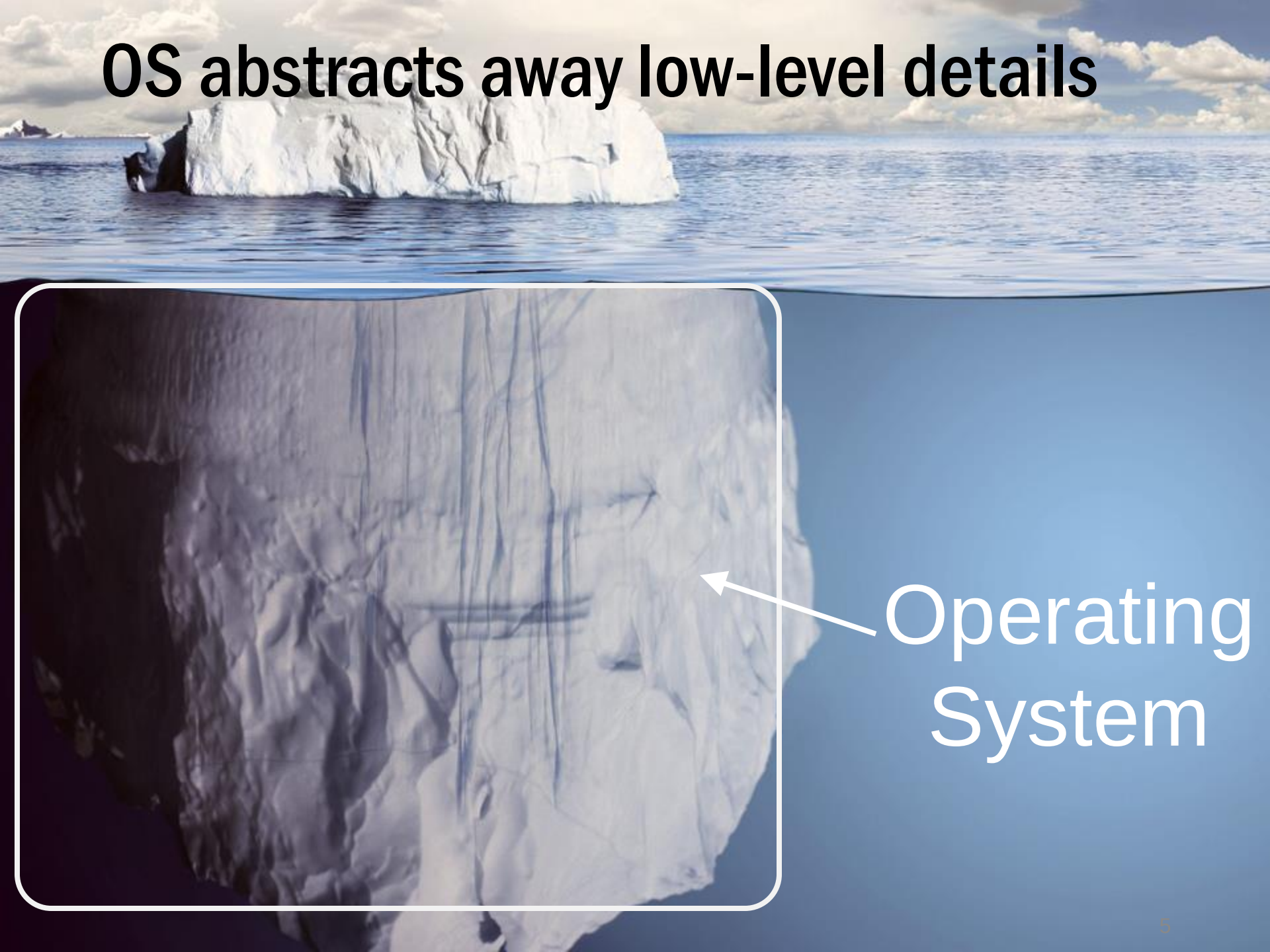
What is an Operating System (OS)?

What is an Operating System (OS)?

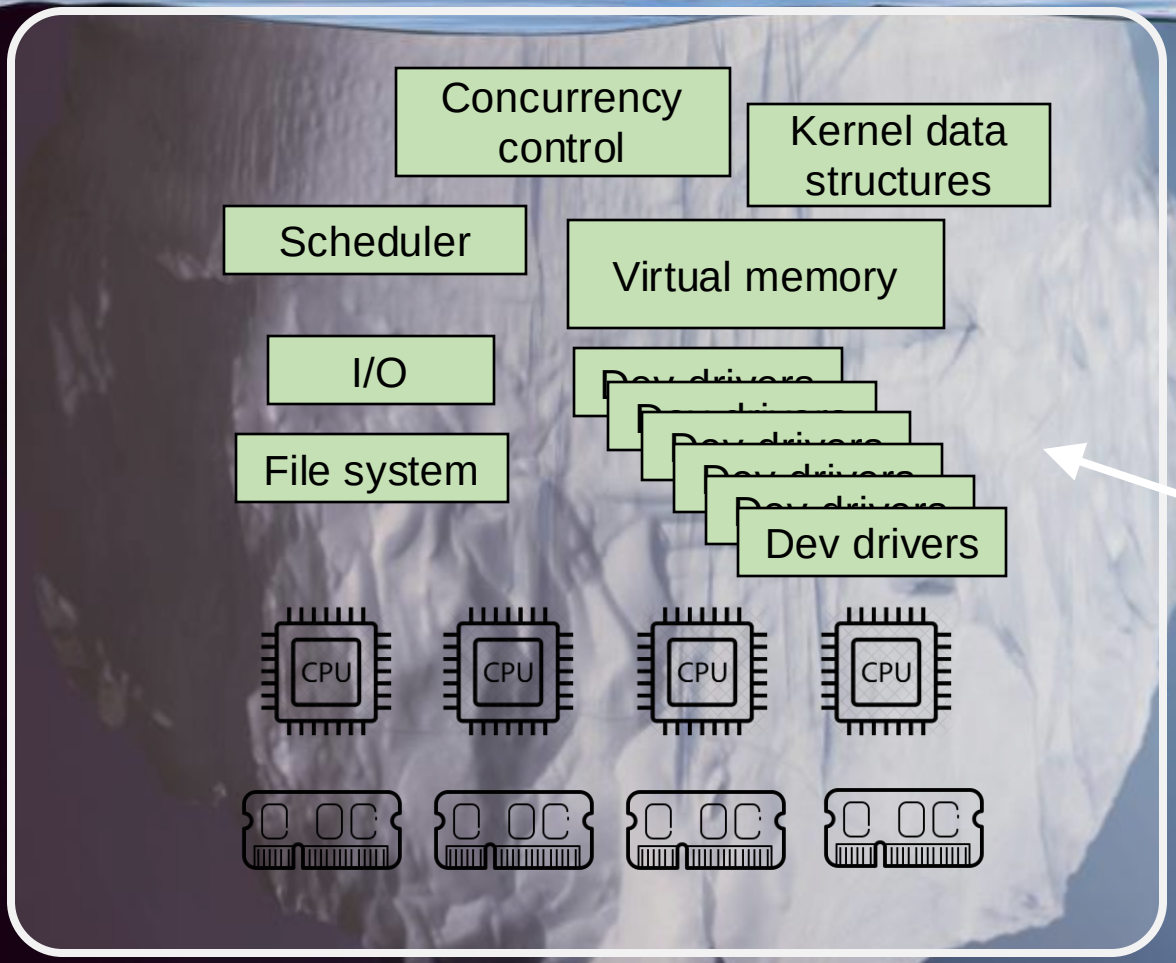
- OS manages resources
  - Memory, CPU, storage, network
  - Data (file systems, I/O)

- Provides low-level abstractions to applications
  - Files
  - Processes, threads
  - Virtual machines (VMs), containers
  - …

OS abstracts away low-level details



OS abstracts away low-level details



Operating System

OS abstracts away low-level details

System call interfaces

Concurrency control

Kernel data structures

Scheduler

Virtual memory

I/O

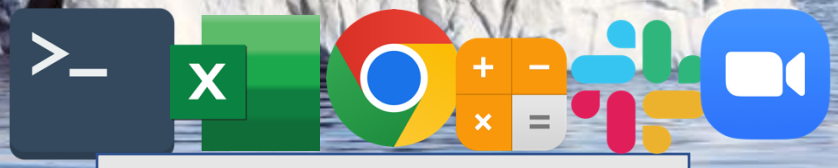
File system

Dev drivers

CPU CPU CPU CPU

Operating System

OS abstracts away low-level details



System call interfaces

Virtualization

Concurrency

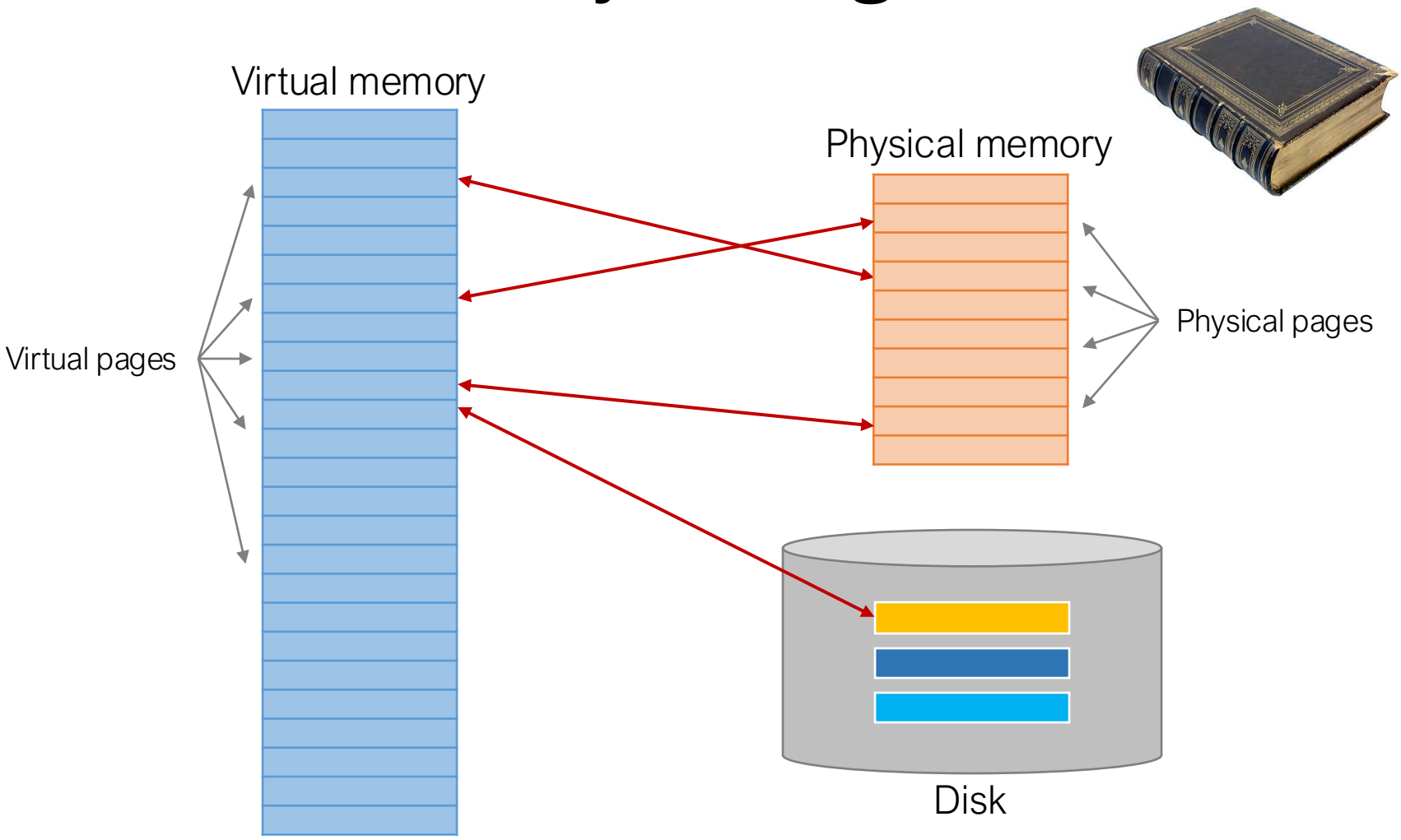
Persistence

Operating System

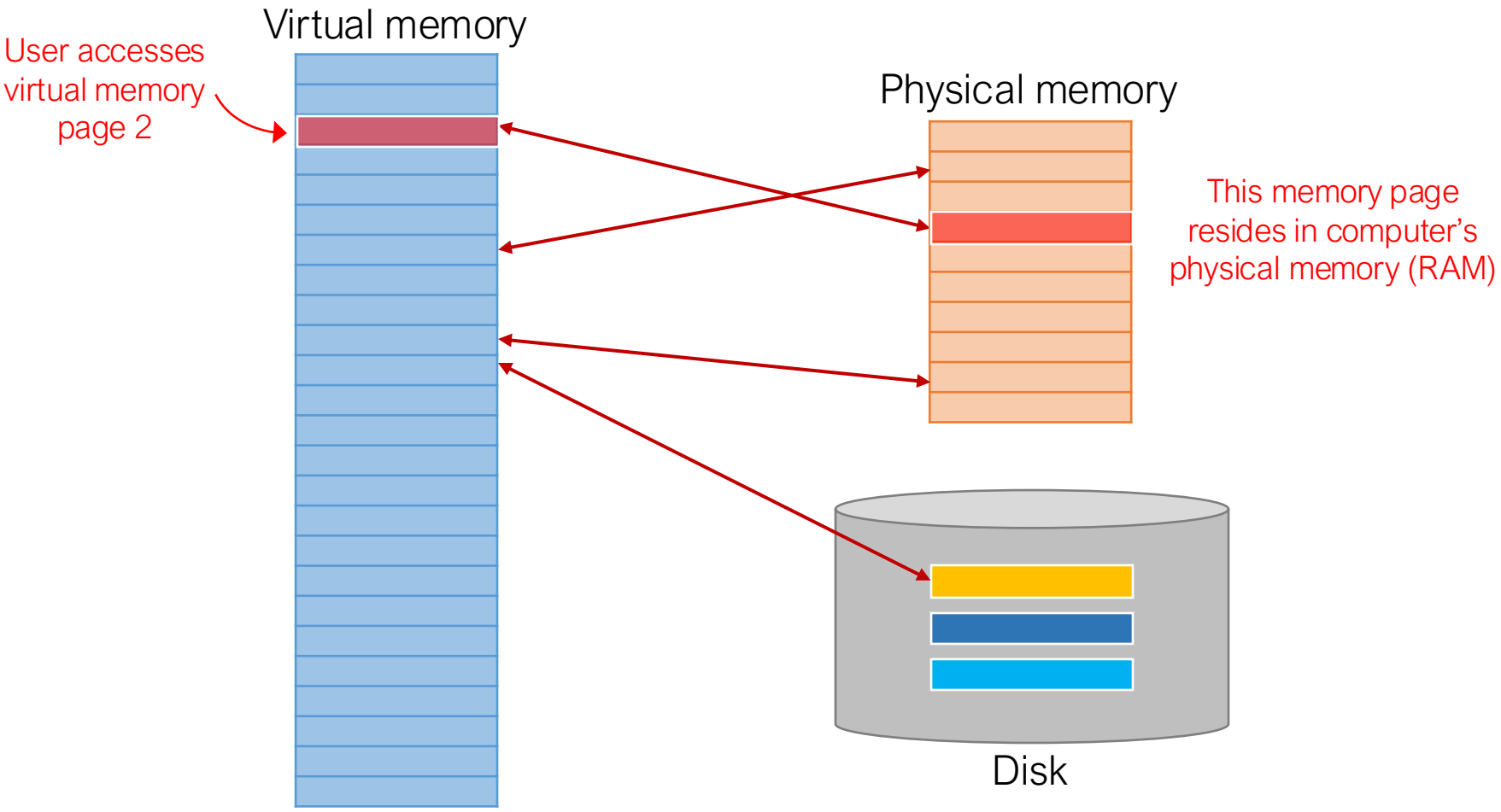
Virtualizing memory

- The physical memory is an array of bytes
- A program keeps (most of) its data in memory
  - Read memory (load): Access an address to fetch the data
  - Write memory (store): Store the data to a given address
- Virtual memory spanning {`RAM + disk`}
  - A virtual memory address is made up (by OS+hardware)
  - All memory addresses seen by human and user apps are virtual

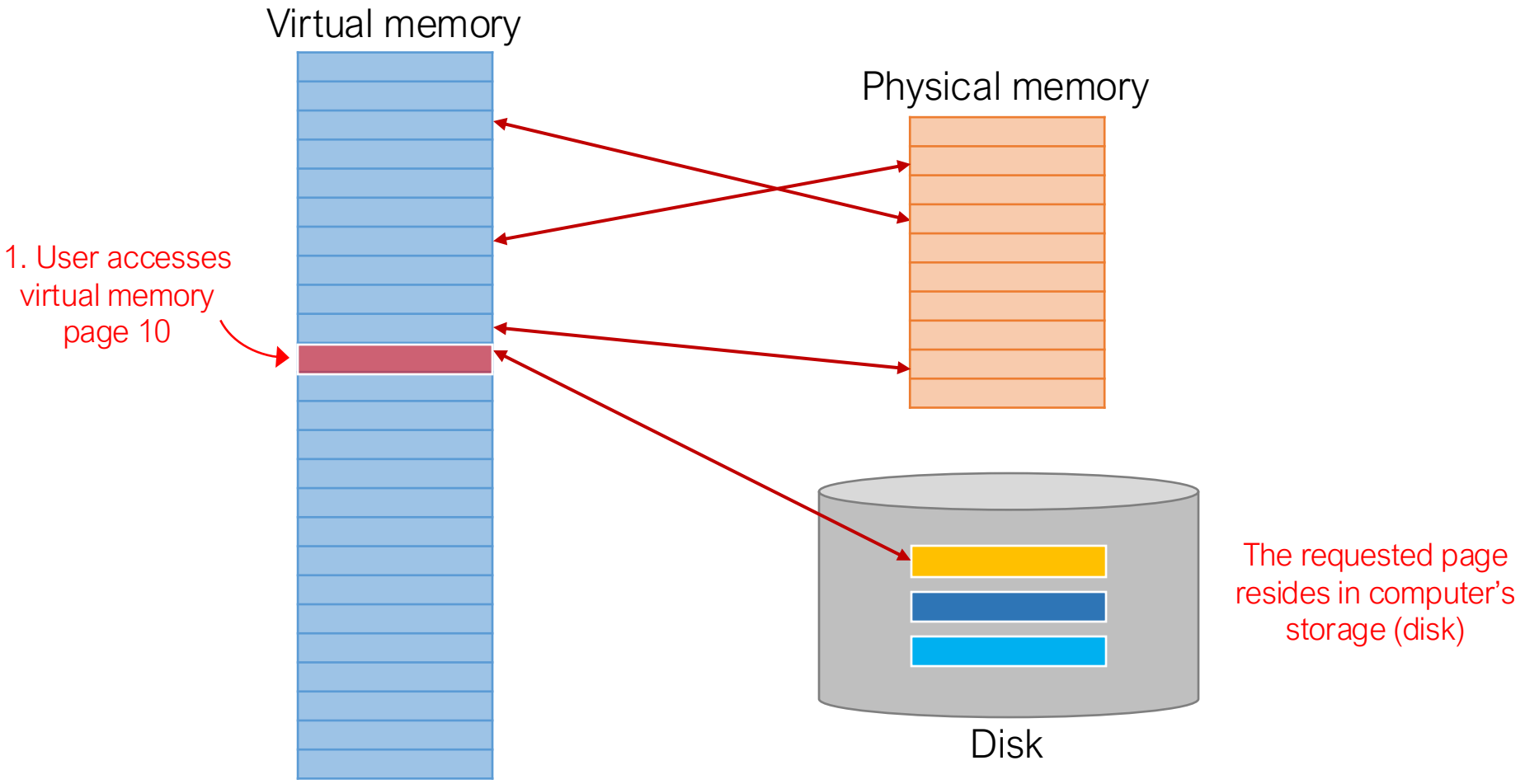
Virtual memory management



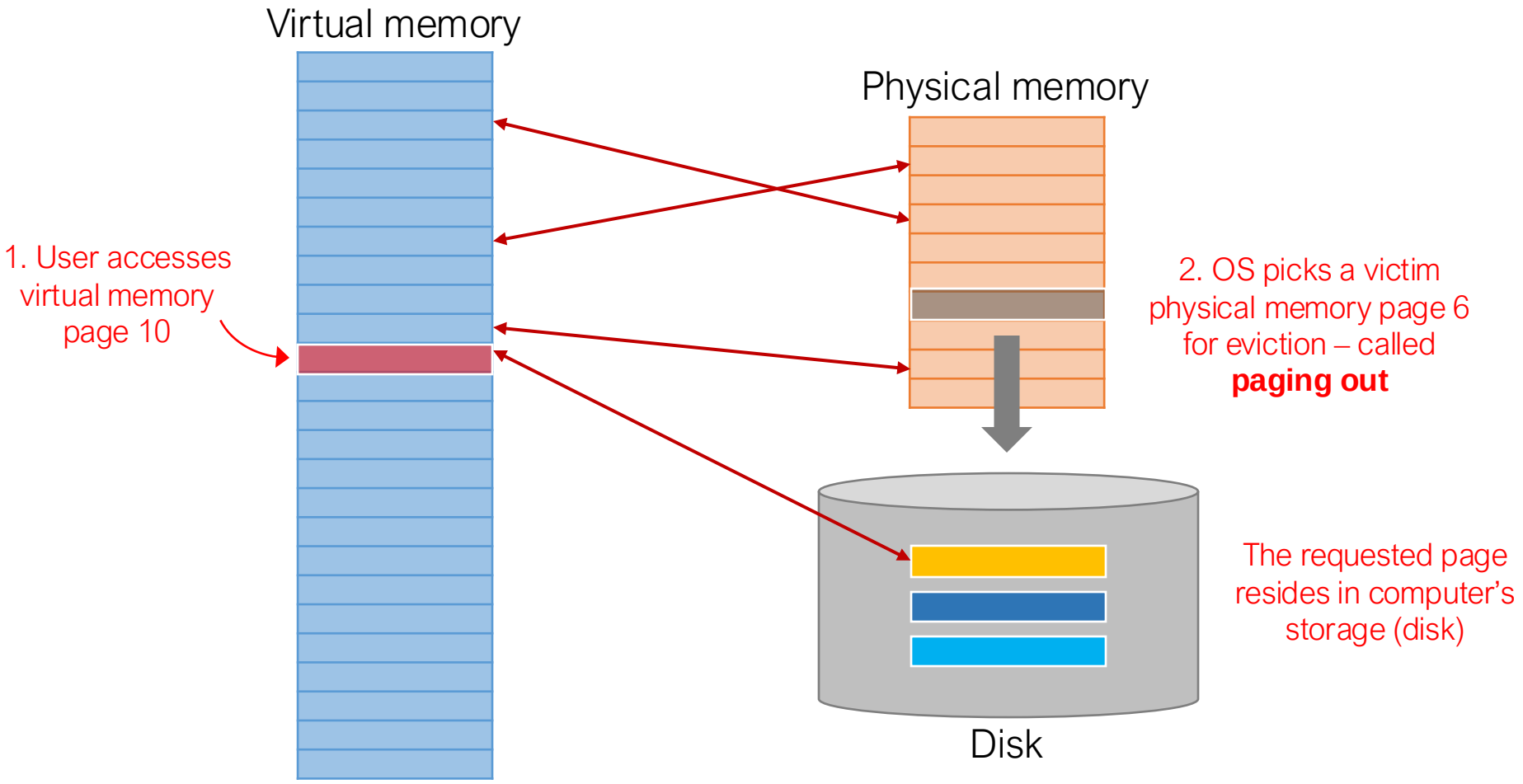
Virtual memory management



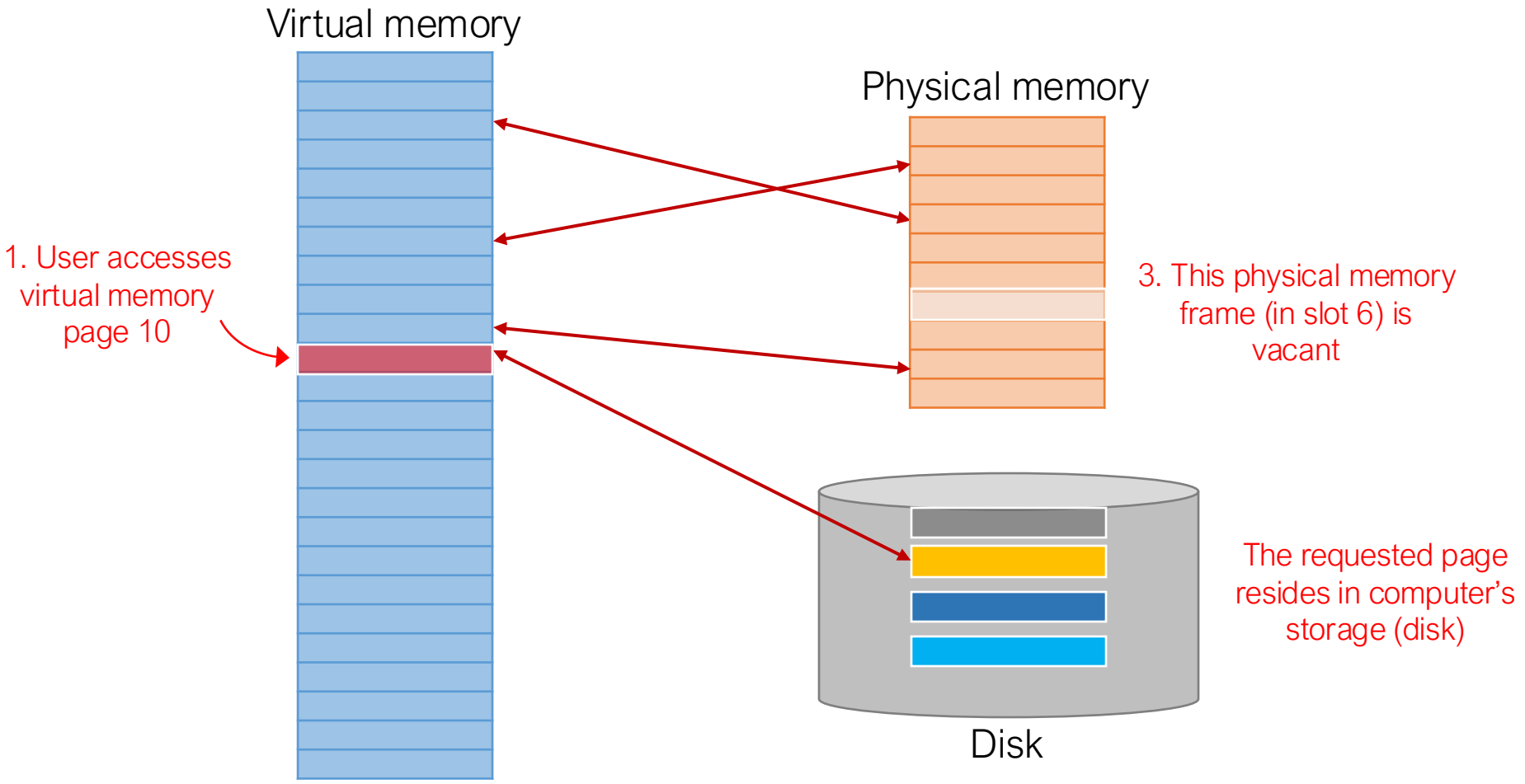
Virtual memory management



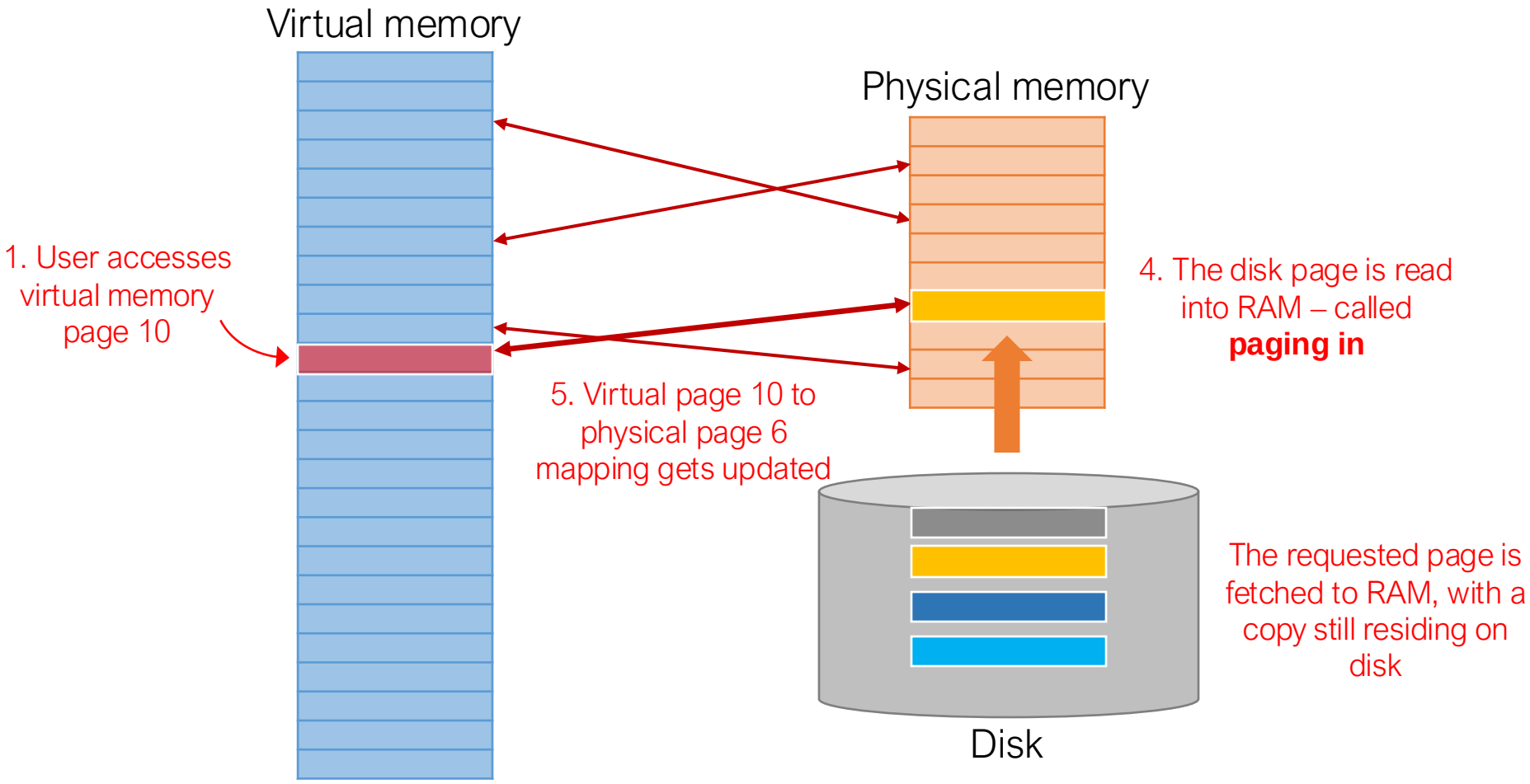
Virtual memory management



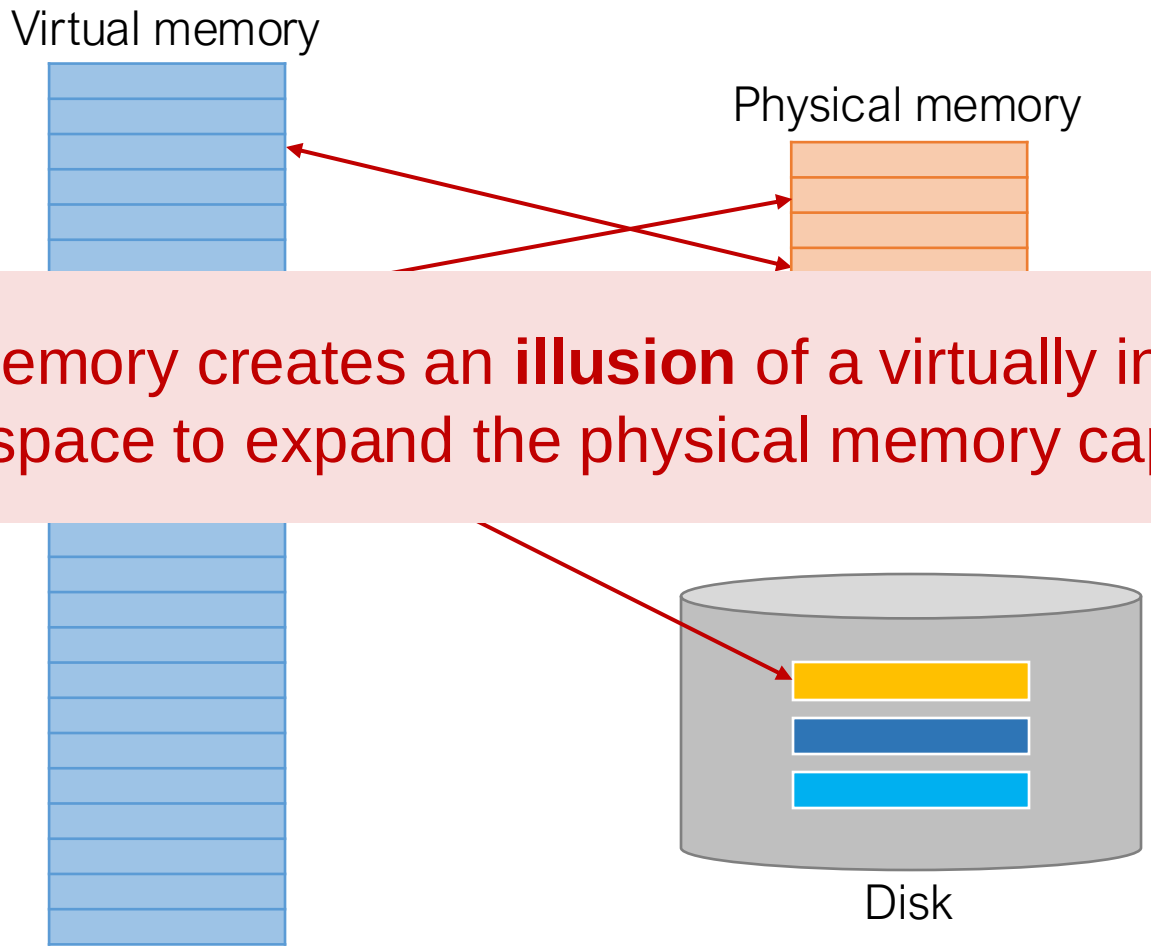
Virtual memory management



Virtual memory management

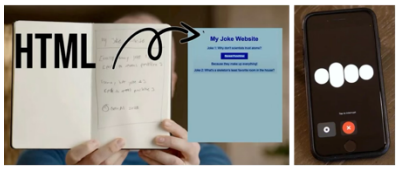


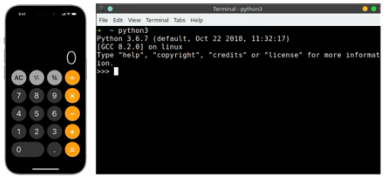
Virtual memory management

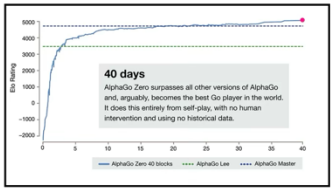


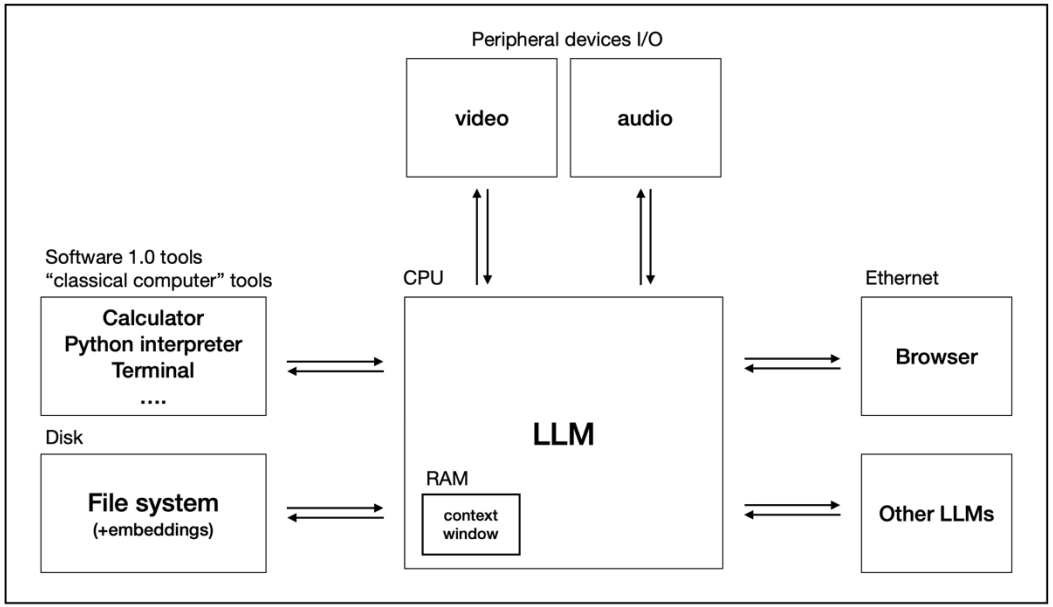
Virtual memory creates an **illusion** of a virtually infinite memory space to expand the physical memory capacity

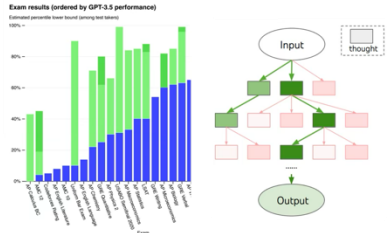
LLMs as Operating Systems (LLM OS)

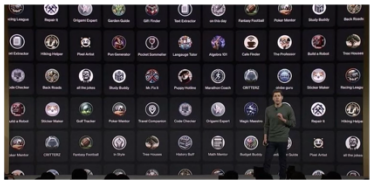












An LLM in a few years:

It can read and generate text
It has more knowledge than any single human about all subjects
It can browse the internet
It can use the existing software infrastructure (calculator, Python, mouse/keyboard)
It can see and generate images and video
It can hear and speak, and generate music
It can think for a long time using a System 2
It can "self-improve" in domains that offer a reward function
It can be customized and finetuned for specific tasks, many versions exist in app stores
It can communicate with other LLMs

* Andrej Karpathy. Intro to LLMs.

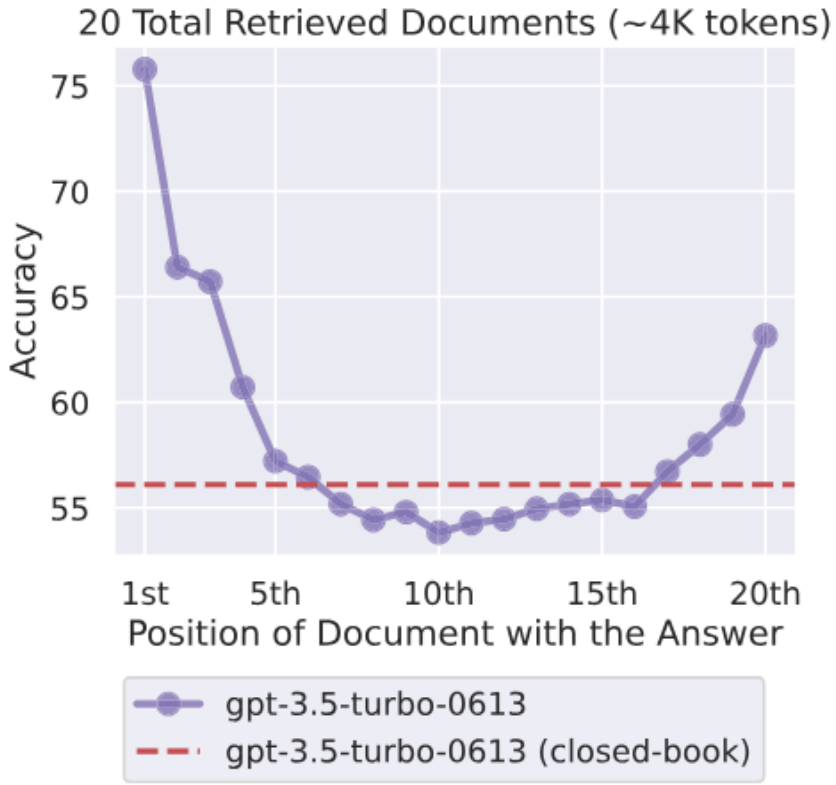
Pain points of long context

- LLMs' context window size is limited (though newest LLMs have dramatically increased this limit)

Model	Open-source?	Token limit
Llama-2	✅	4096
Llama-3	✅	4096
GPT-3.5-turbo	❌	16385
GPT-4o	❌	128000
Claude3.5-Sonnet	❌	200000
Claude3.7-Sonnet	❌	128000

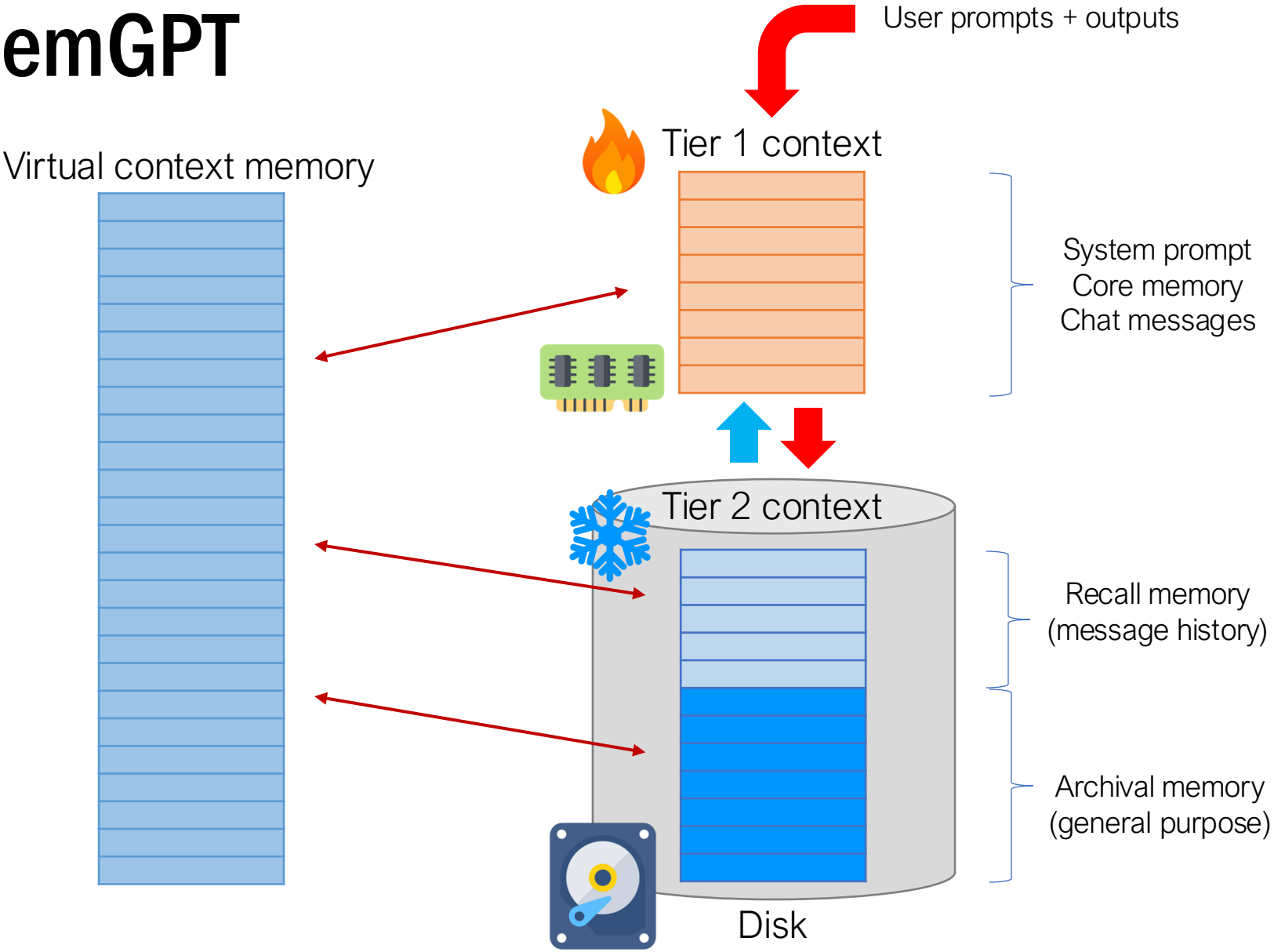
Pain points of long context

- Long context easily loses focus
  - generates irrelevant and redundant information
  - “lost in the middle”

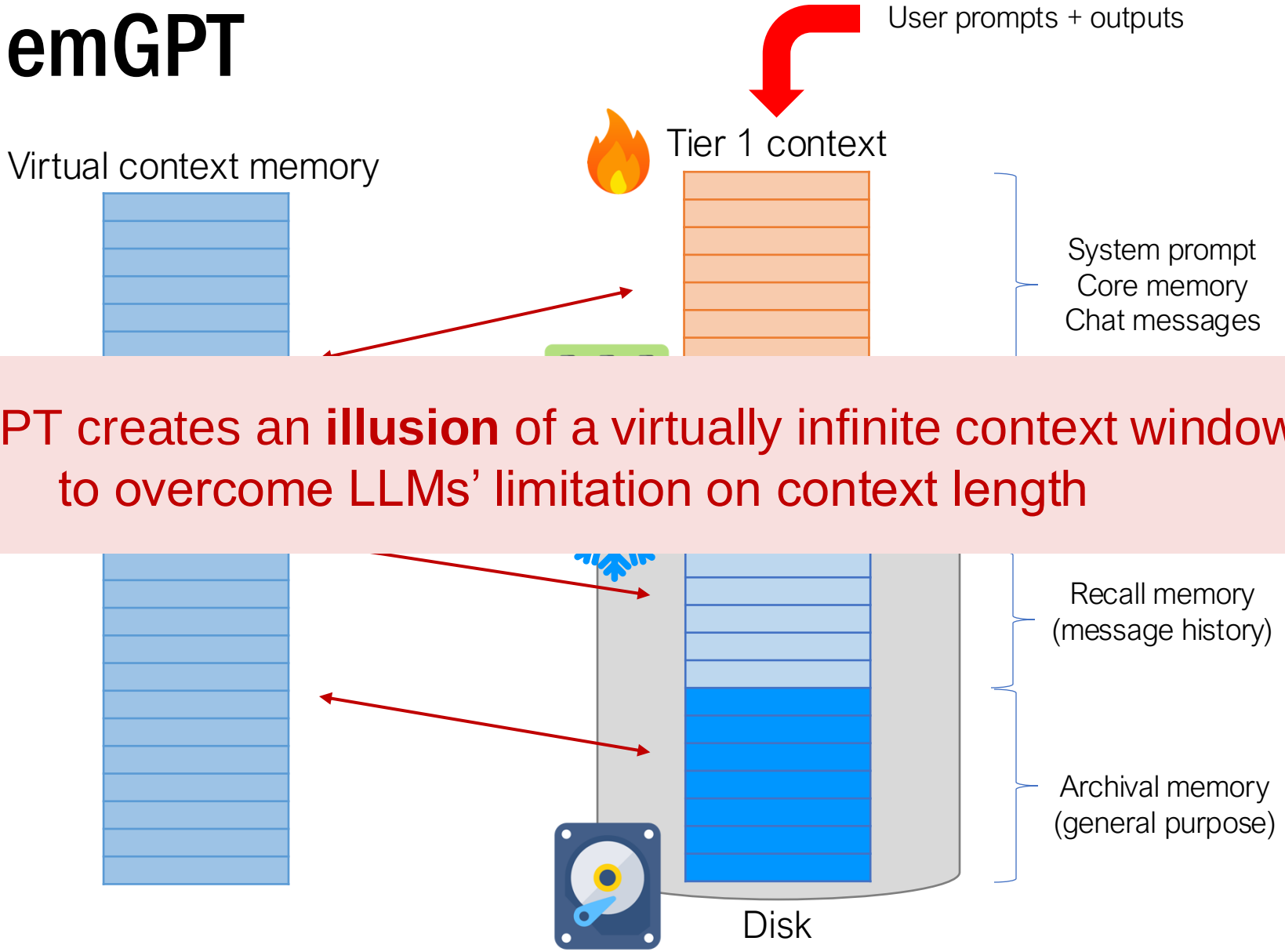


* Liu et. al. Lost in the Middle: How Language Models Use Long Contexts

MemGPT

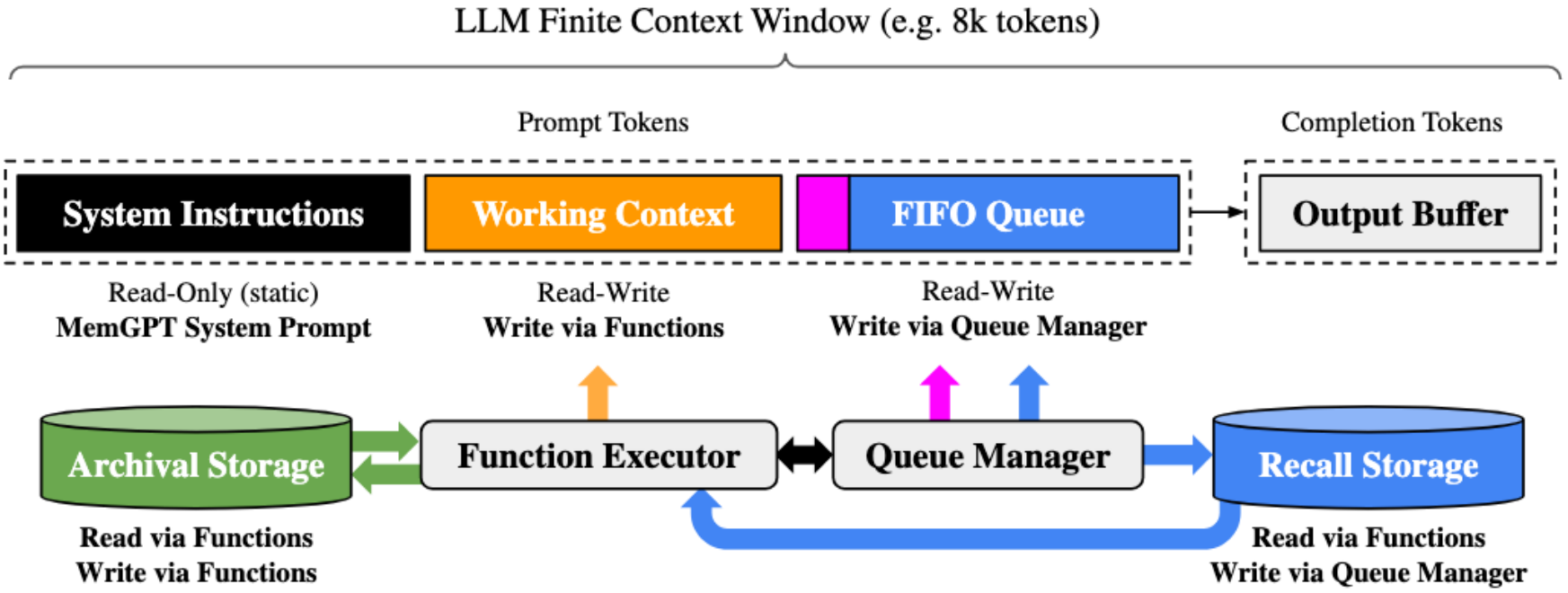

User prompts + outputs
Virtual context memory
Tier 1 context
System prompt
Core memory
Chat messages
Tier 2 context
Recall memory
(message history)
Archival memory
(general purpose)
Disk

MemGPT



MemGPT creates an **illusion** of a virtually infinite context window to overcome LLMs' limitation on context length

MemGPT workflow



Tier 1 context

Tier 2 context

Functions are tools

Recall storage stores summarized context of archival

MemGPT demo