Sistemas Operativos

LEI - 2019/2020

:: Estrutura dos Sistemas Operativos ::

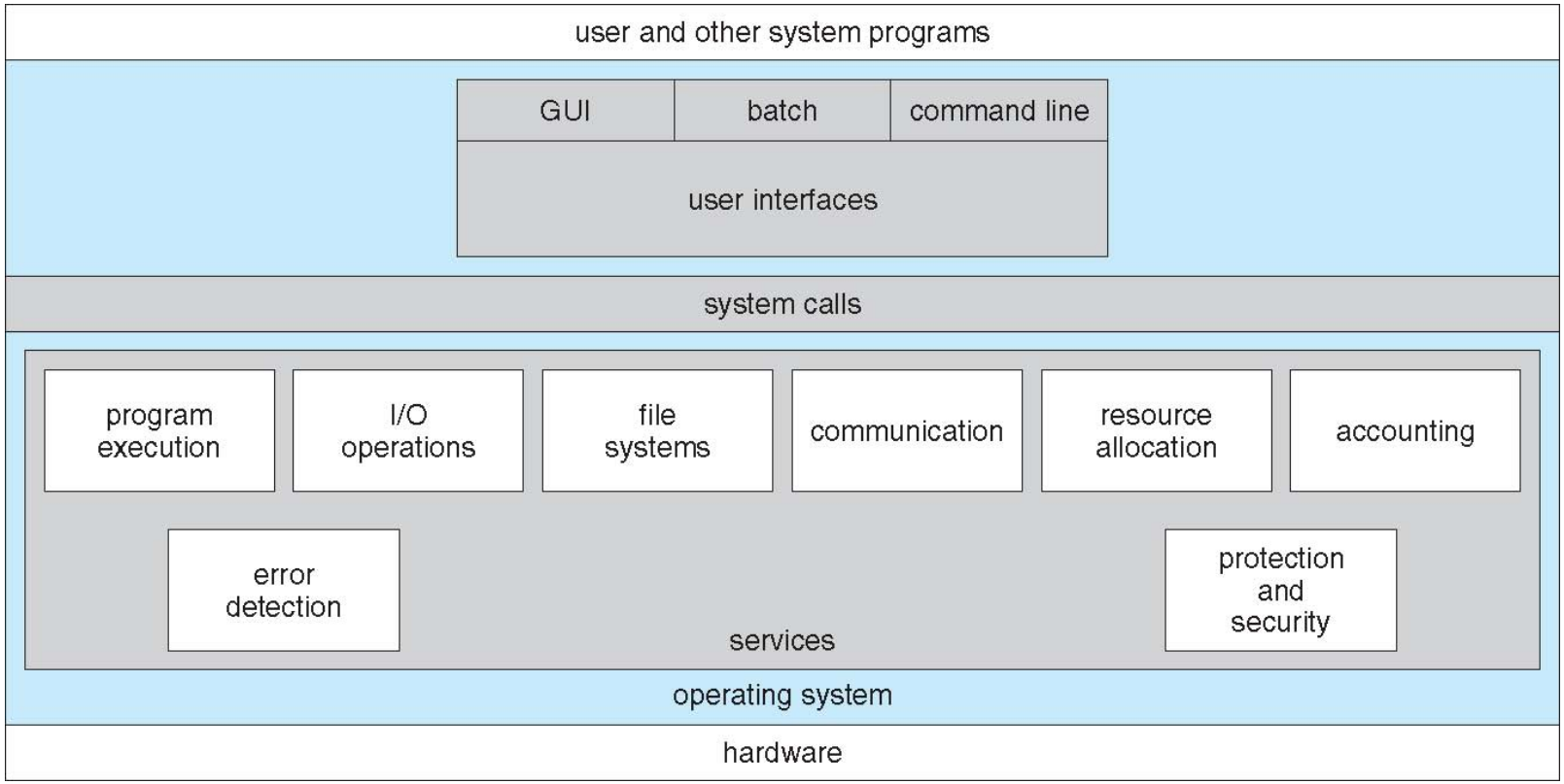
Escola Superior de Tecnologia de Setúbal - IPS

Conteúdos

- Serviços de um Sistema Operativo
- Estrutura de um Sistema Operativo
- Processo de arranque (bootloader)

Serviços de um SO

- Ambiente para execução de programas
- Alguns serviços relevantes:
  - interface de utilizador (shell, gui, etc.)
  - execução de programas
  - operações de i/o
  - manipulação de ficheiros
  - comunicações
  - detecção de erros
  - etc.

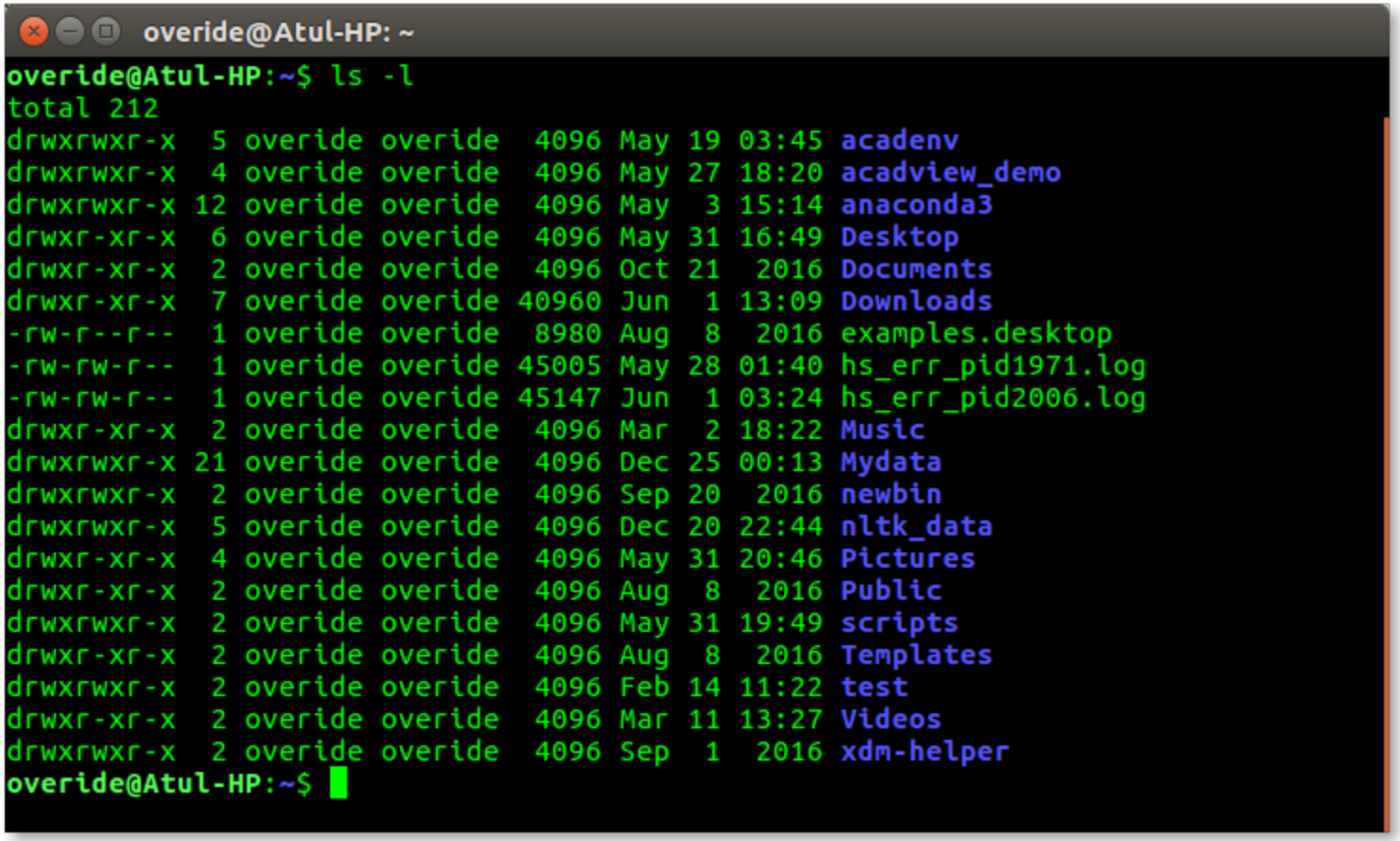

user and other system programs
GUI
batch
command line
user interfaces
system calls
program execution
I/O operations
file systems
communication
resource allocation
accounting
error detection
protection and security
services
operating system
hardware

Interface com o Sistema Operativo

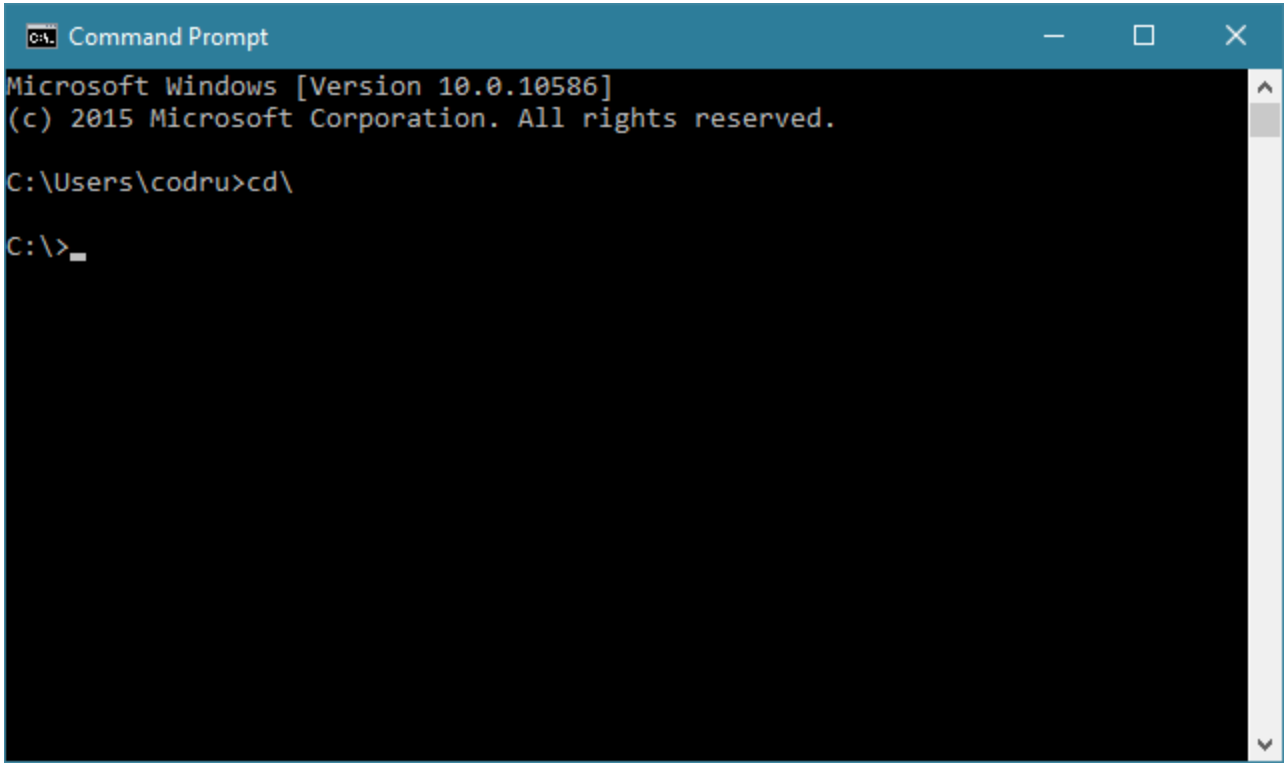
CLI (Command-Line Interpreter - shell)

GUI (Graphical User Interface)

Ex: bash (Bourne-again shell)

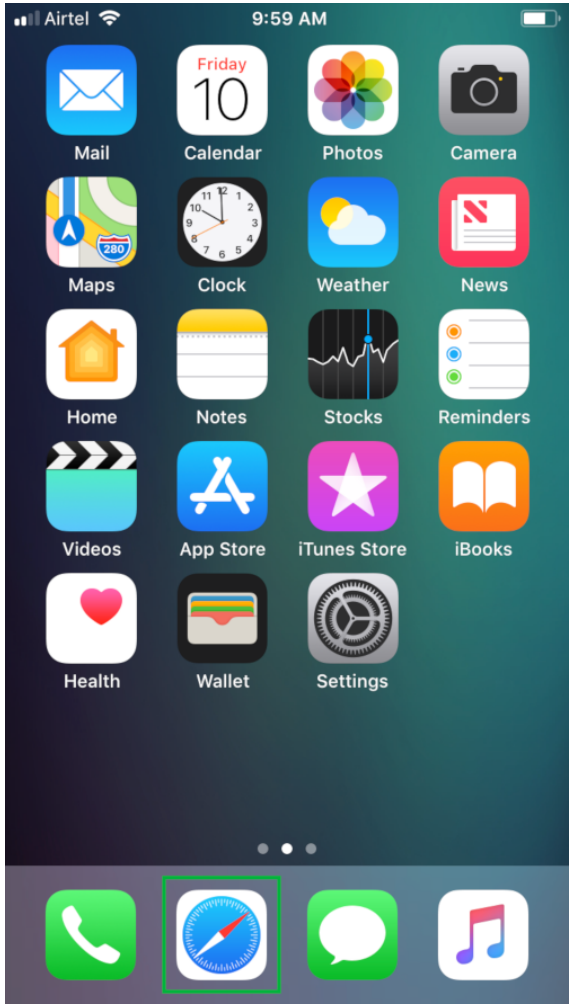


Ex: Windows command prompt



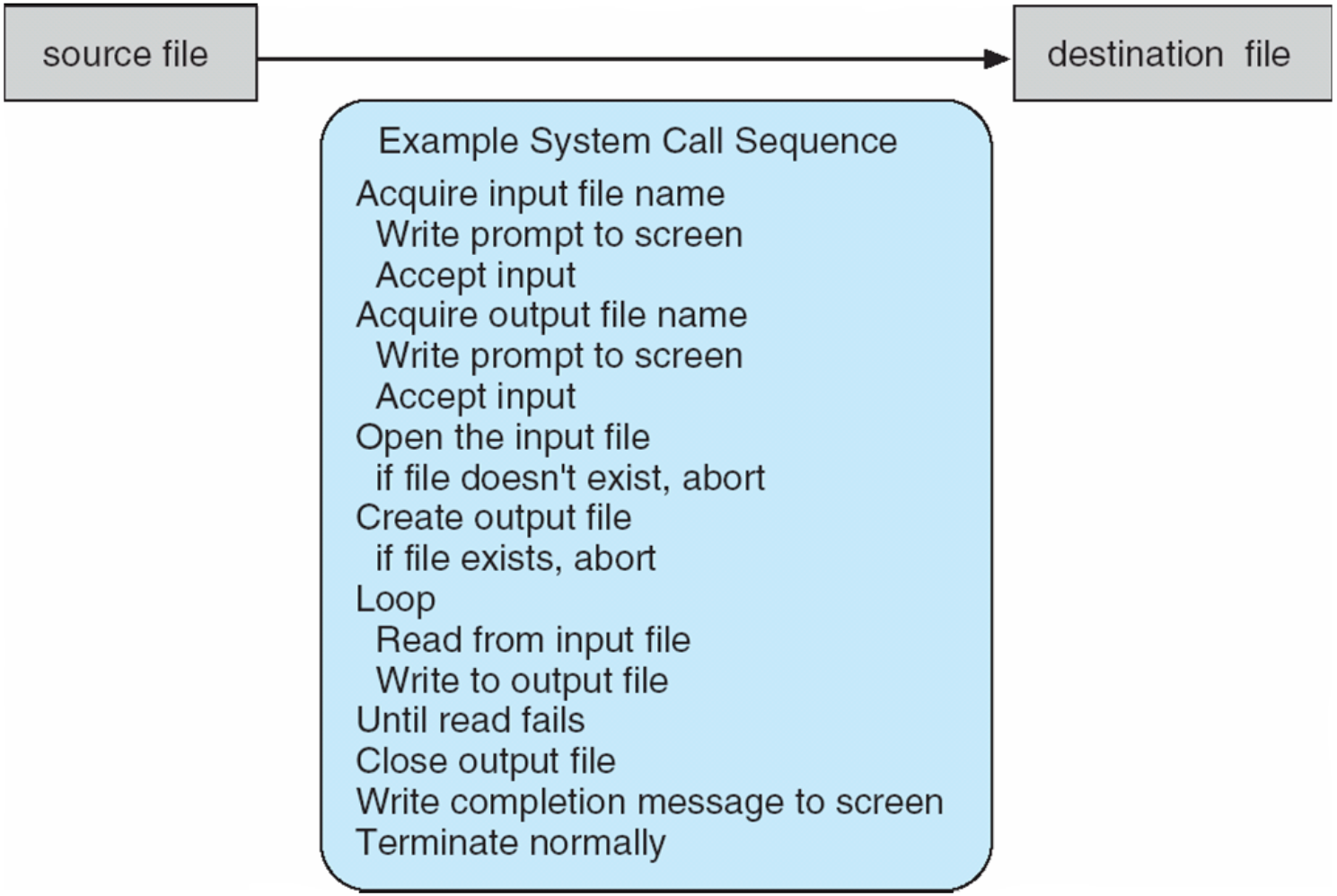
Ex: macOS Desktop

Ex: iPhone touchscreen



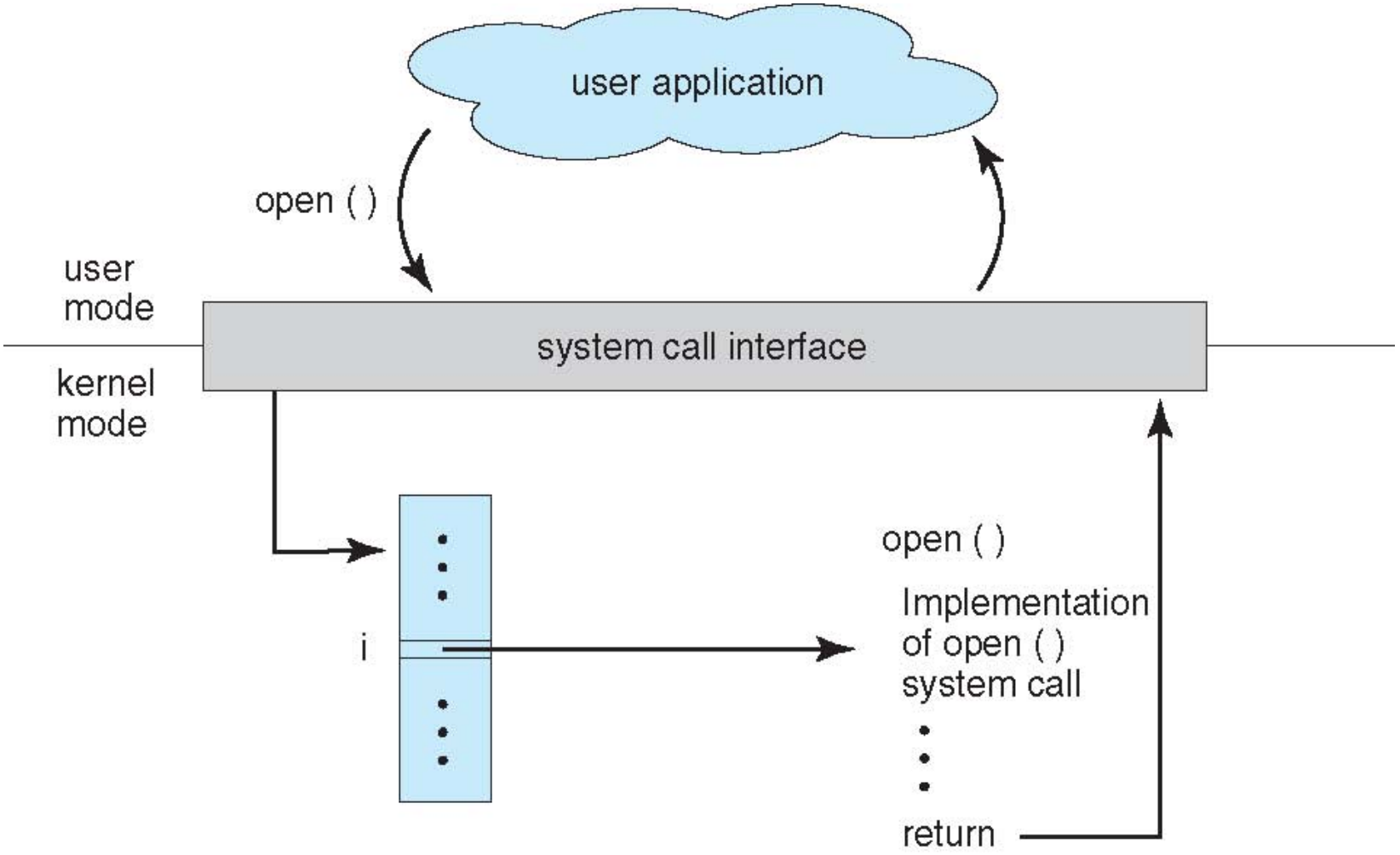
System calls

- Interface com os serviços do SO
- Geralmente implementado em C/C++/Asm


source file
destination file
Example System Call Sequence
Acquire input file name
Write prompt to screen
Accept input
Acquire output file name
Write prompt to screen
Accept input
Open the input file
if file doesn't exist, abort
Create output file
if file exists, abort
Loop
Read from input file
Write to output file
Until read fails
Close output file
Write completion message to screen
Terminate normally

Syscalls vs API

- APIs abstraem alguns detalhes das syscalls
- Ex APIs:
  - Win32 (createFile, ReadFile, etc.)
  - POSIX (read, write, etc..)
  - Java API (System.out.println, etc.)


user application
open ( )
user mode
kernel mode
system call interface
i
open ( )
Implementation of open ( ) system call
return

Linux: open() syscall

```
SYSCALL_DEFINE3(open, const char __user *, filename,
                int, flags, umode_t, mode)
{
    if (force_o_largefile())
        flags |= O_LARGEFILE;

    return do_sys_open(AT_FDCWD, filename, flags, mode);
}
```

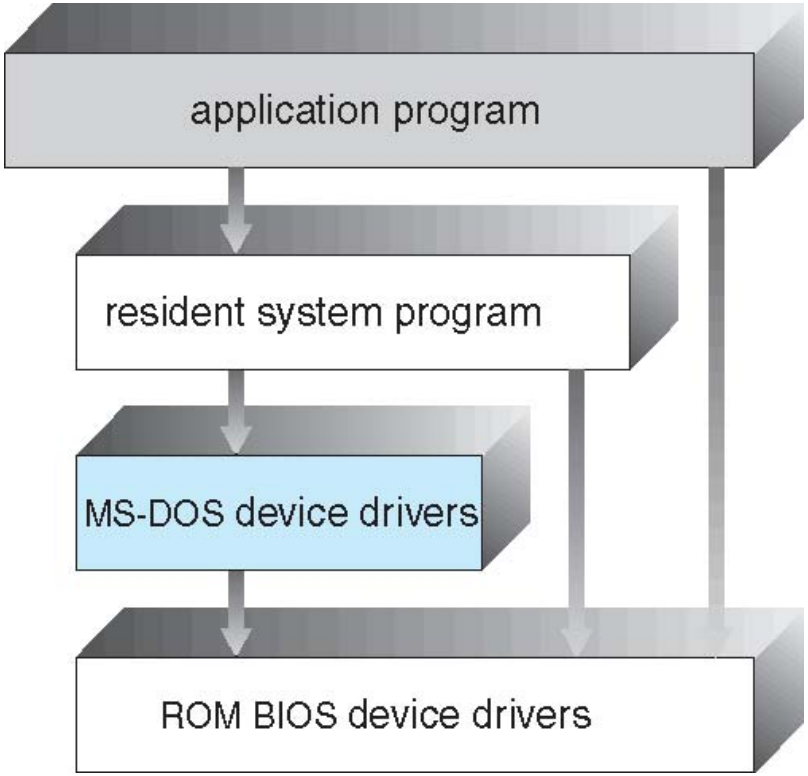
Tipos de syscalls

	Windows	Unix
Process Control	CreateProcess() ExitProcess() WaitForSingleObject()	fork() exit() wait()
File Manipulation	CreateFile() ReadFile() WriteFile() CloseHandle()	open() read() write() close()
Device Manipulation	SetConsoleMode() ReadConsole() WriteConsole()	ioctl() read() write()
Information Maintenance	GetCurrentProcessID() SetTimer() Sleep()	getpid() alarm() sleep()
Communication	CreatePipe() CreateFileMapping() MapViewOfFile()	pipe() shmget() mmap()
Protection	SetFileSecurity() InitlializeSecurityDescriptor() SetSecurityDescriptorGroup()	chmod() umask() chown()

Estruturas de SO

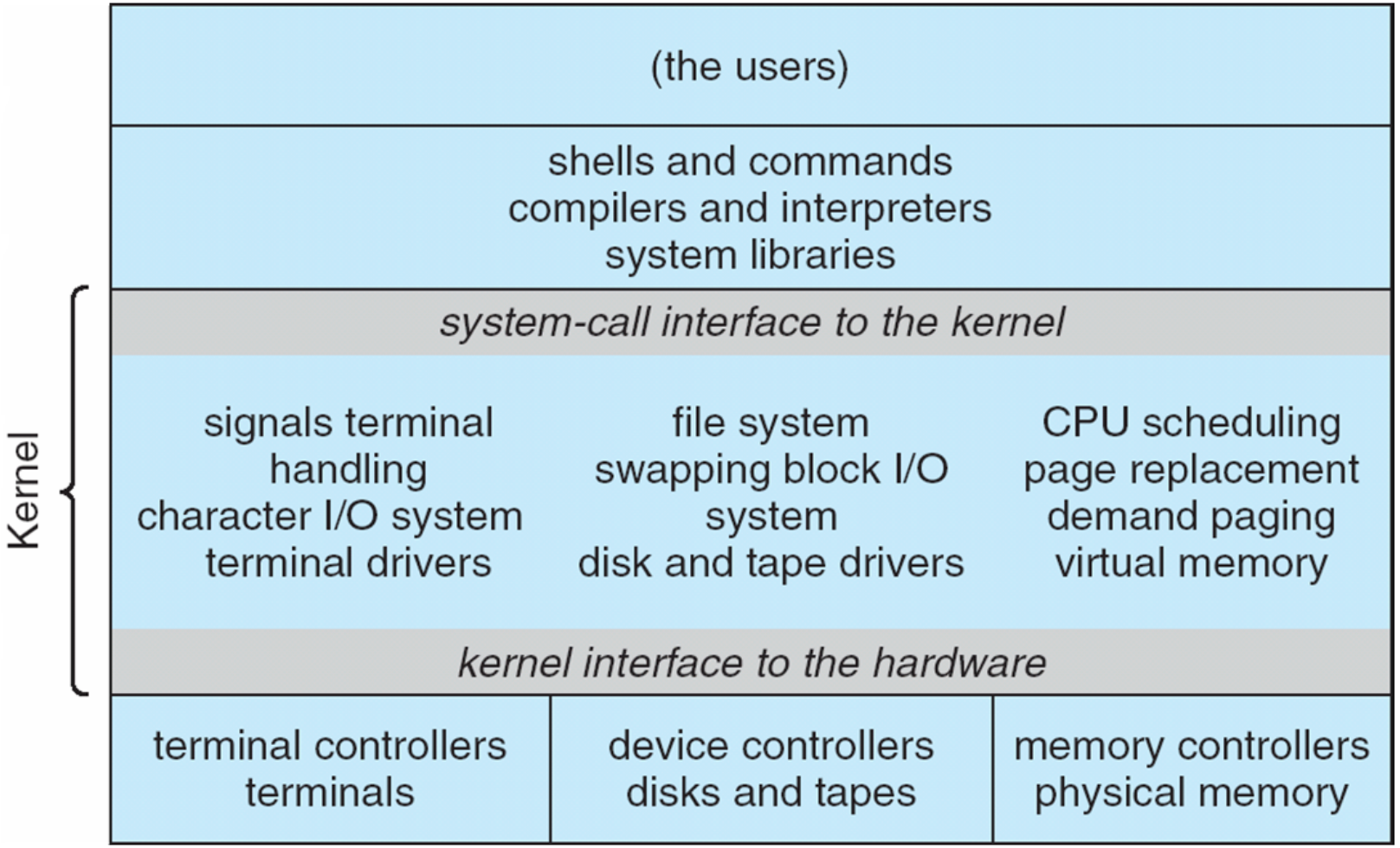
- Simples
- Camadas
- Microkernel
- Módulos
- Hibridos

Estrutura simples (ex: MS-DOS)

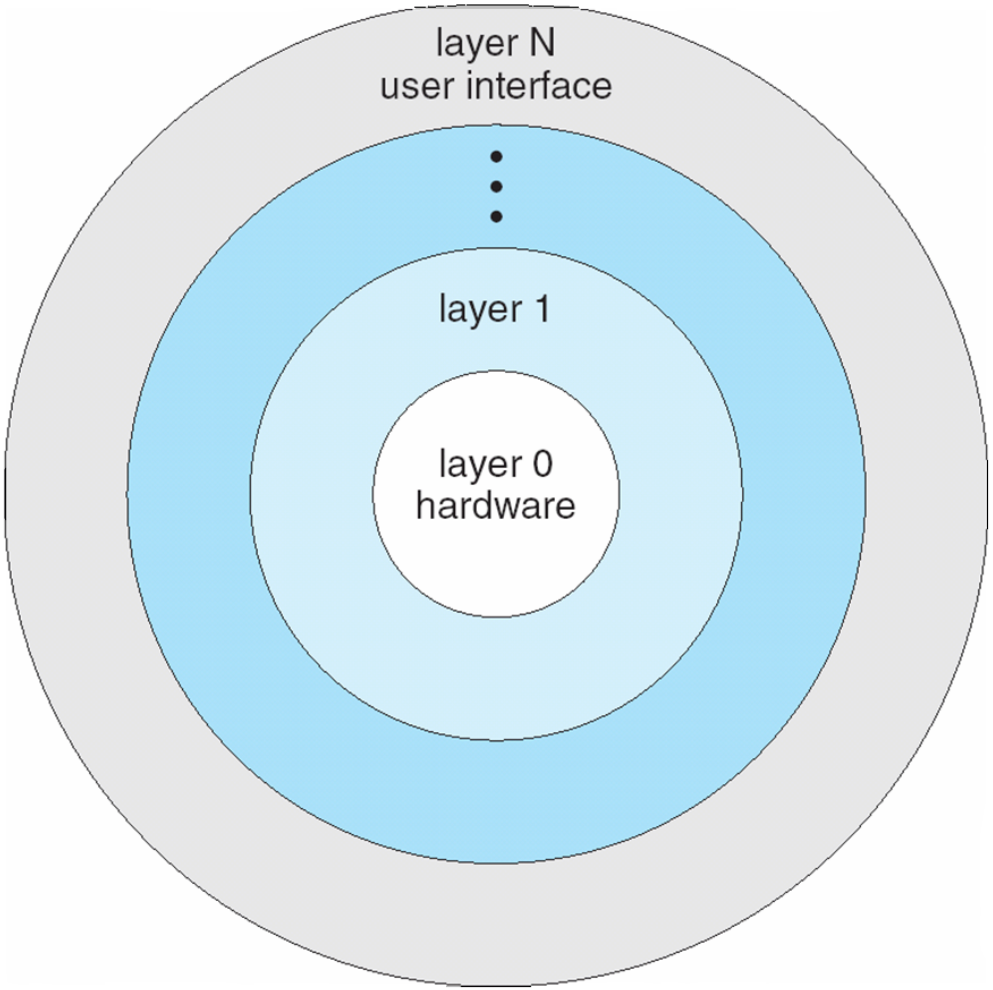


⚠️ Sem divisão em módulos..

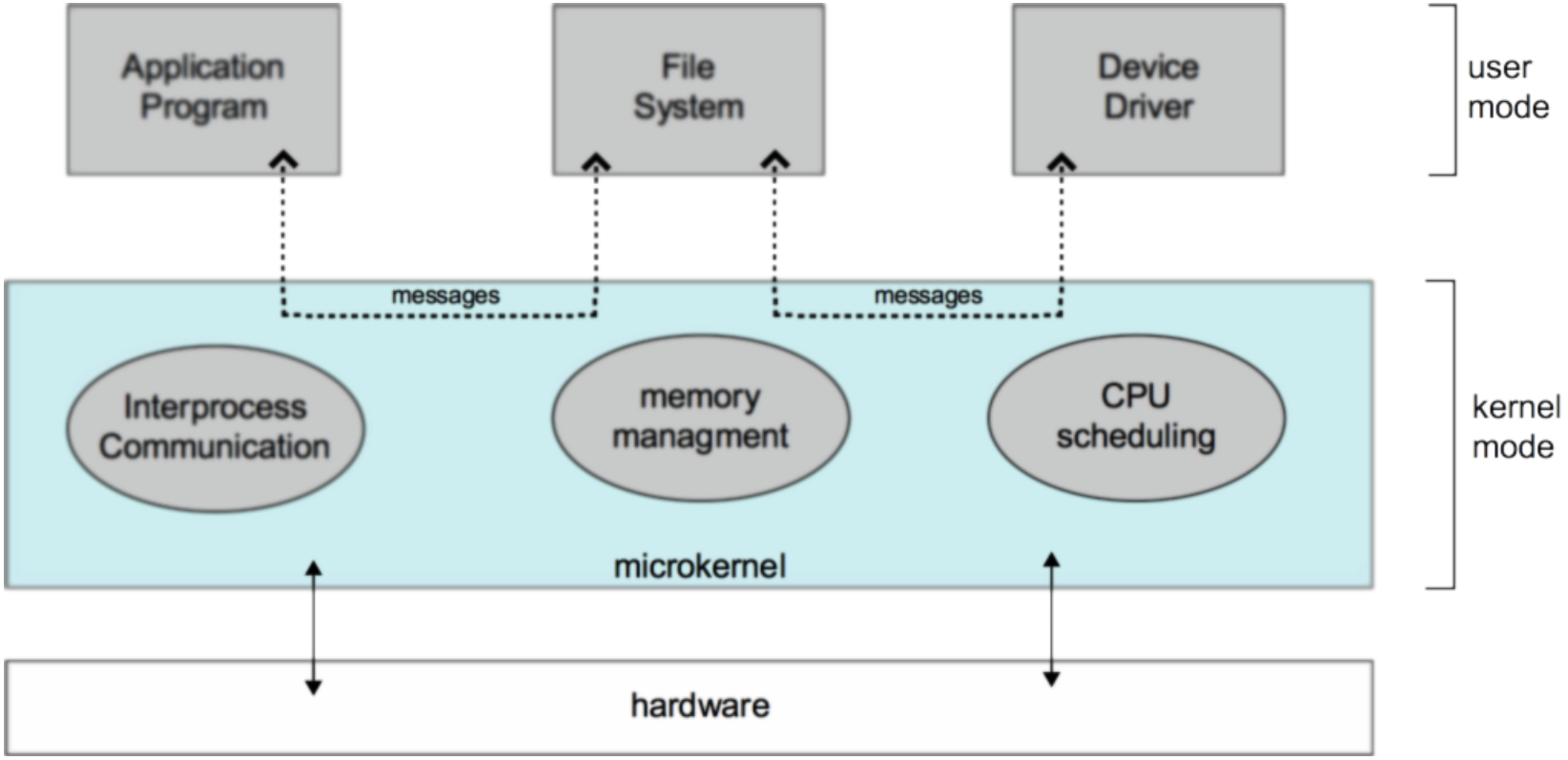
Estrutura não-tão simples (ex: UNIX)



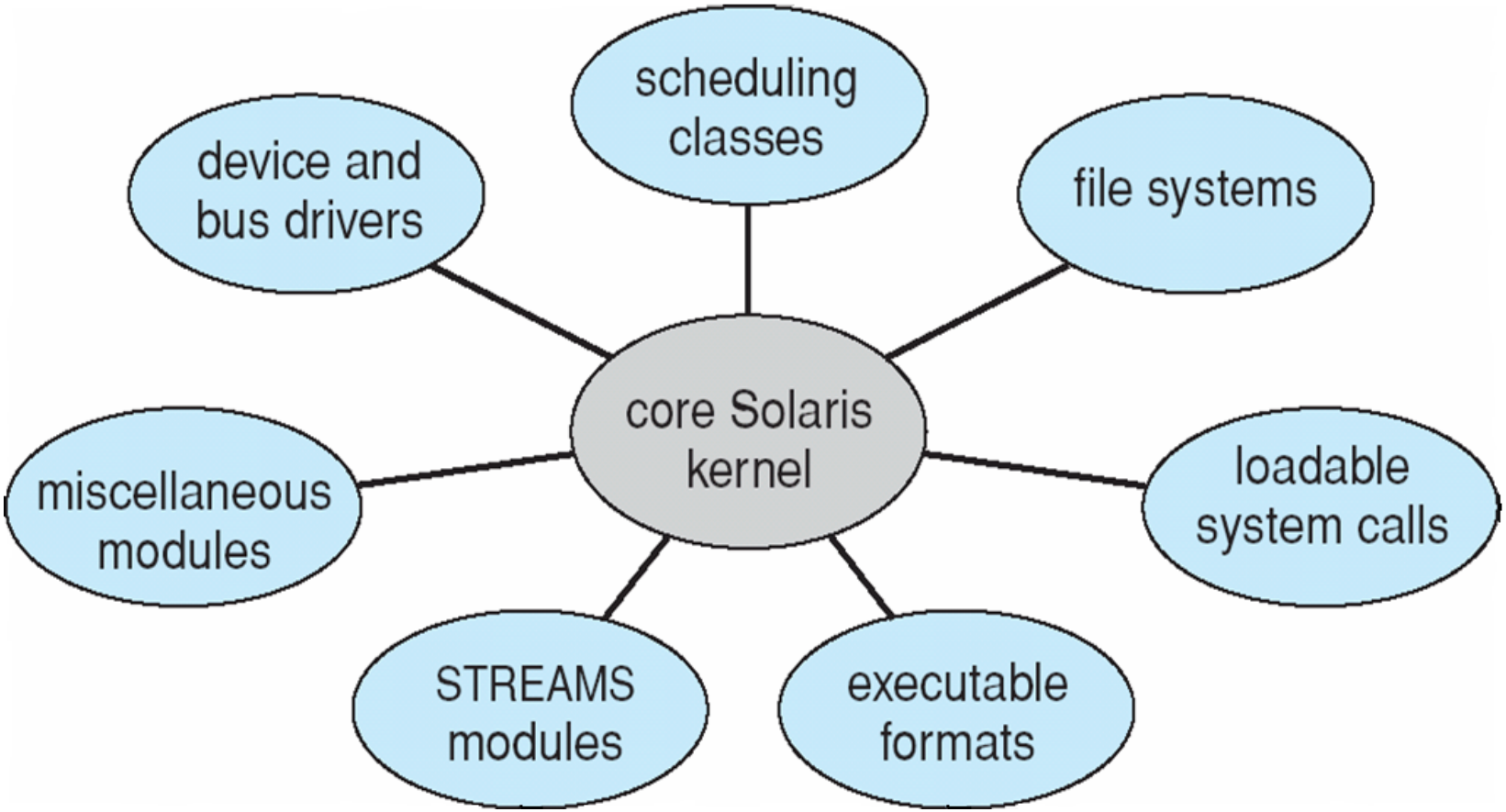
Estrutura camadas



Estrutura microkernel (ex: minix)



Estrutura modular (ex: Solaris)

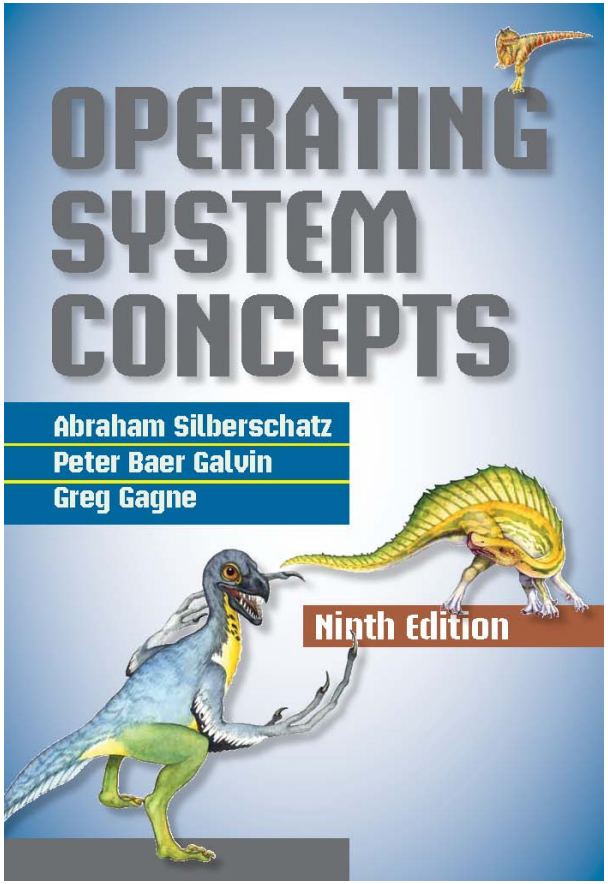


Quiz...

System boot...

Sumário

- Sistemas operativos fornecem serviços
- Num nível mais baixo, serviços são fornecidos através de syscalls ou APIs
- Num nível mais alto, serviços são fornecidos via shells/GUIs
- Vários serviços (gestão de processos, ficheiros, etc.)
- Vários tipos de estrutura (SO são programas complexos)



Ler capítulo 2...