HPC


НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ЯДЕРНЫЙ УНИВЕРСИТЕТ
МИФИ

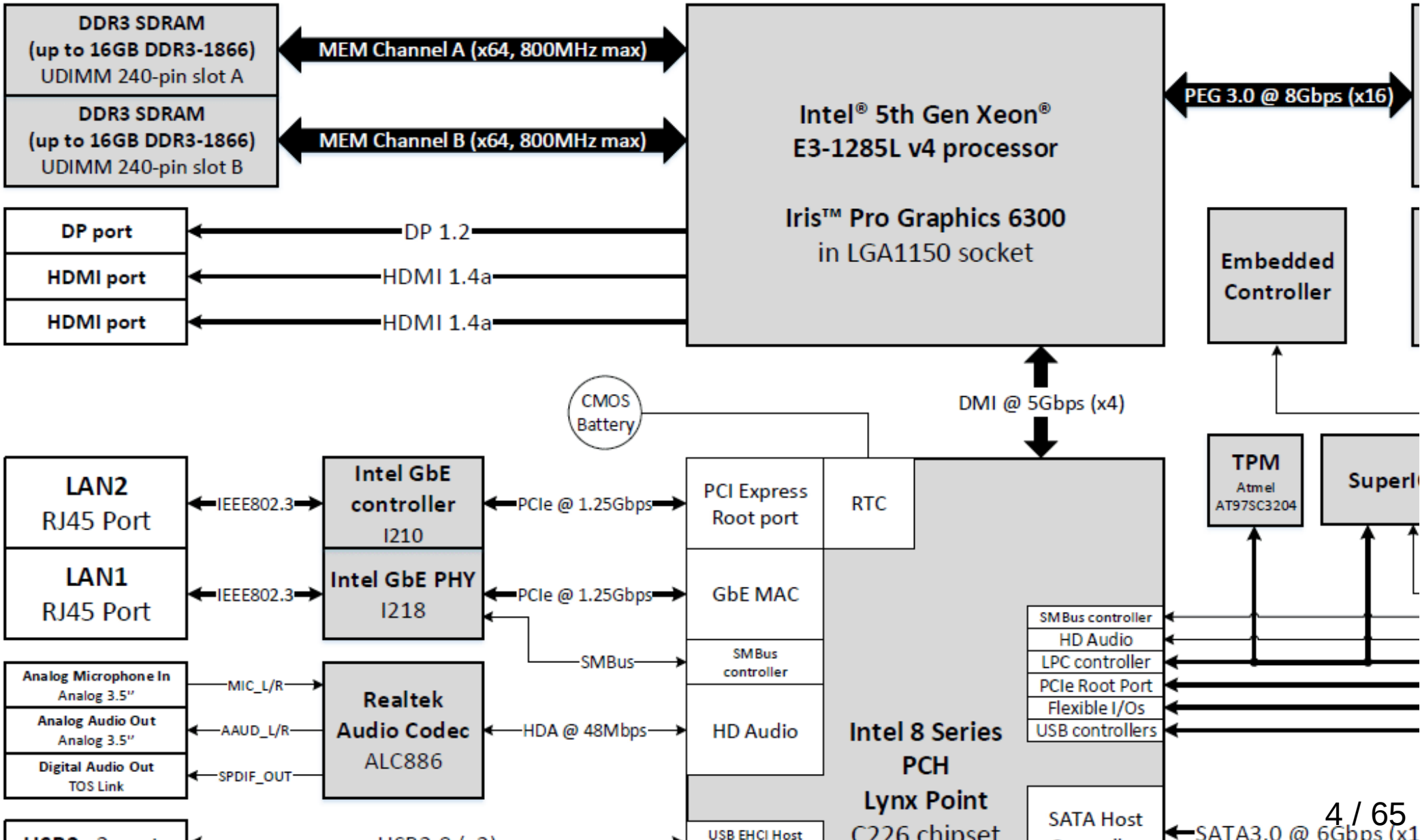
Основные характеристики вычислительного кластера

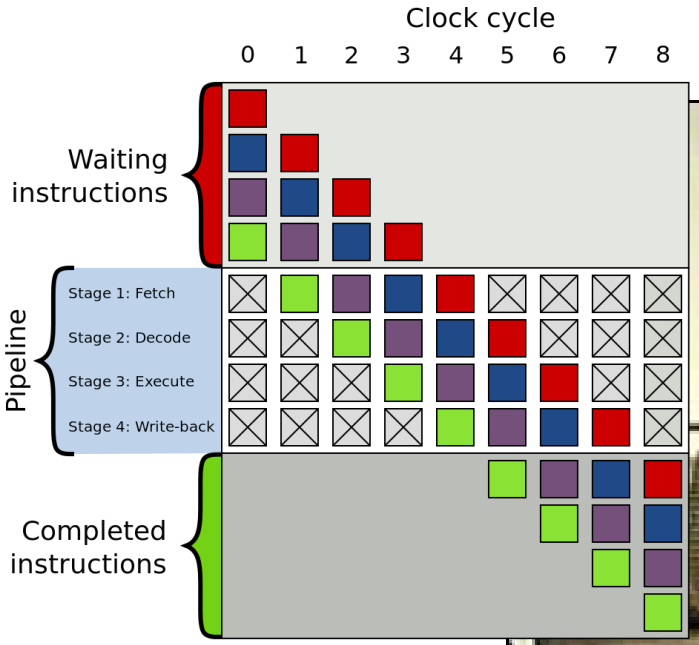
- Пиковая (теоретическая) производительность (Tflops)
- Linpack производительность (Tflops)
- BogoMIPS производительность?
- Пропускная способность Interconnect (Gbps)
- Задержки Interconnect (ns)

Основные характеристики вычислительного кластера

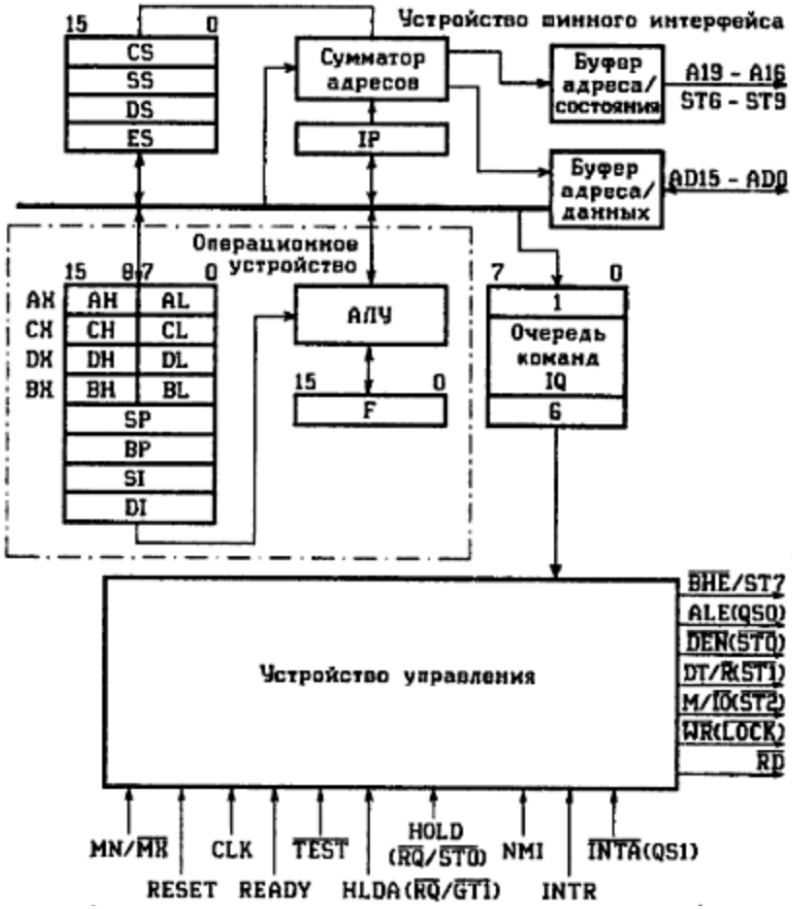
- Пиковая (теоретическая) производительность (Tflops)
- Linpack производительность (Tflops)
- BogoMIPS производительность?
- Пропускная способность Interconnect (Gbps)
- Задержки Interconnect (ns)

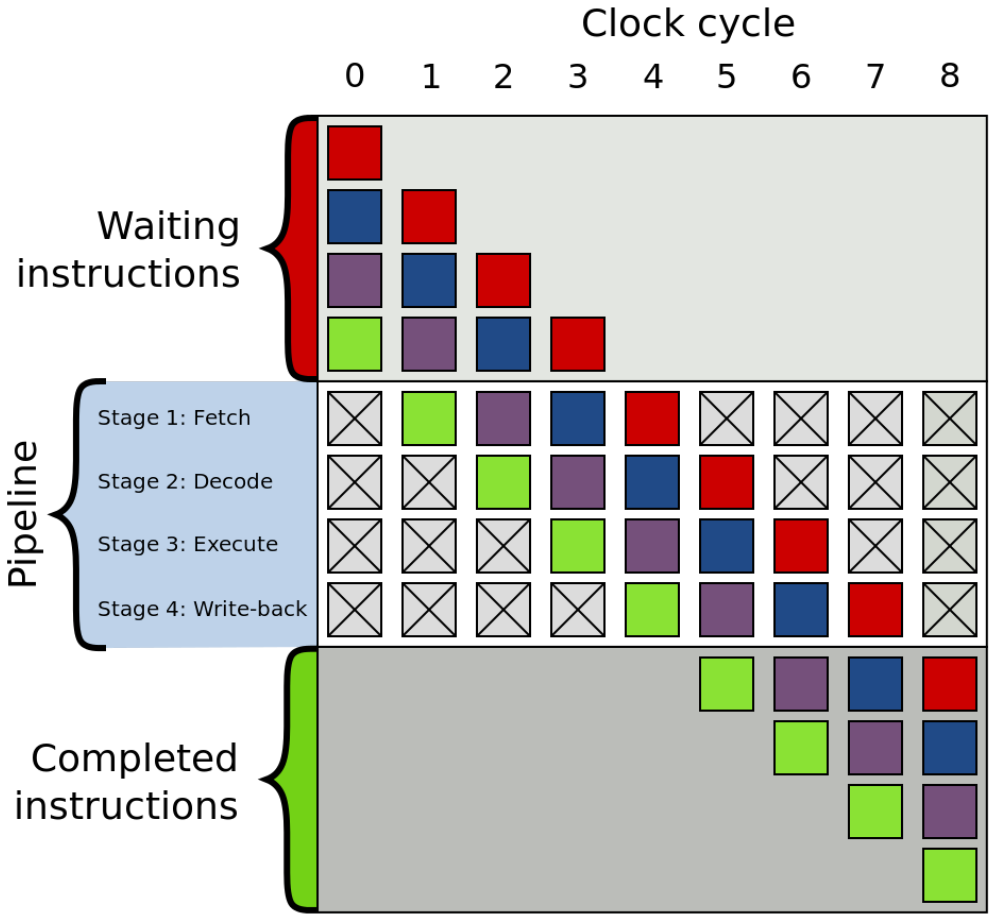
Remember?


DDR3 SDRAM
(up to 16GB DDR3-1866)
UDIMM 240-pin slot A
DDR3 SDRAM
(up to 16GB DDR3-1866)
UDIMM 240-pin slot B
MEM Channel A (x64, 800MHz max)
MEM Channel B (x64, 800MHz max)
Intel® 5th Gen Xeon®
E3-1285L v4 processor
Iris™ Pro Graphics 6300
in LGA1150 socket
PEG 3.0 @ 8Gbps (x16)
DP port
HDMI port
HDMI port
DP 1.2
HDMI 1.4a
HDMI 1.4a
Embedded Controller
CMOS Battery
DMI @ 5Gbps (x4)
TPM
Atmel
AT97SC3204
LAN2
RJ45 Port
LAN1
RJ45 Port
IEEE802.3
IEEE802.3
Intel GbE controller
I210
Intel GbE PHY
I218
PCIe @ 1.25Gbps
PCIe @ 1.25Gbps
SMBus
PCI Express Root port
RTC
GbE MAC
SMBus controller
HD Audio
Analog Microphone In
Analog 3.5"
Analog Audio Out
Analog 3.5"
Digital Audio Out
TOS Link
MIC_L/R
AAUD_L/R
SPDIF_OUT
Realtek Audio Codec
ALC886
HDA @ 48Mbps
Intel 8 Series PCH
Lynx Point
C226 chipset
SMBus controller
HD Audio
LPC controller
PCIe Root Port
Flexible I/Os
USB controllers
SATA Host
USB EHCI Host
SATA3.0 @ 6Gbps (x1



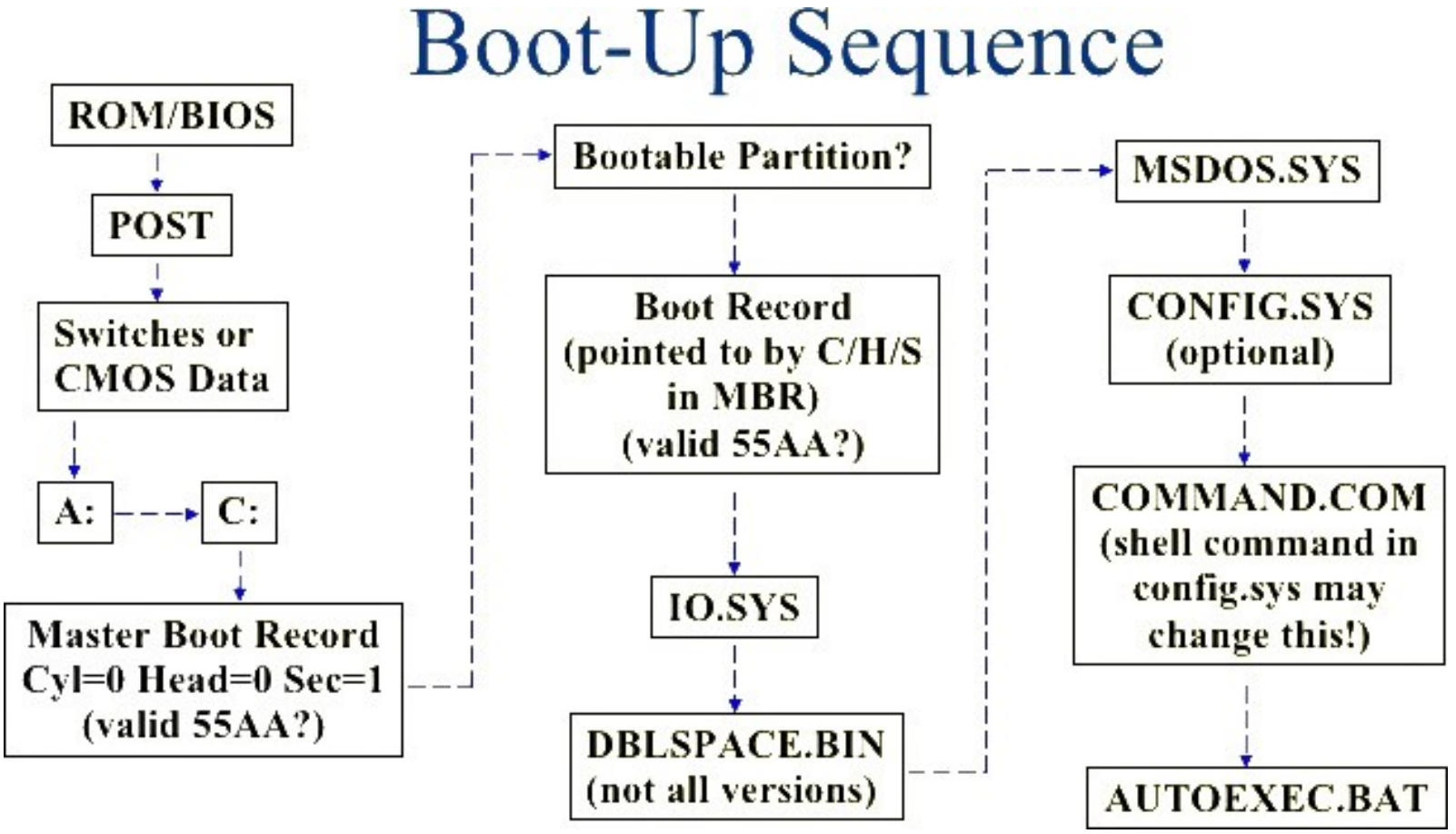
Waiting instructions

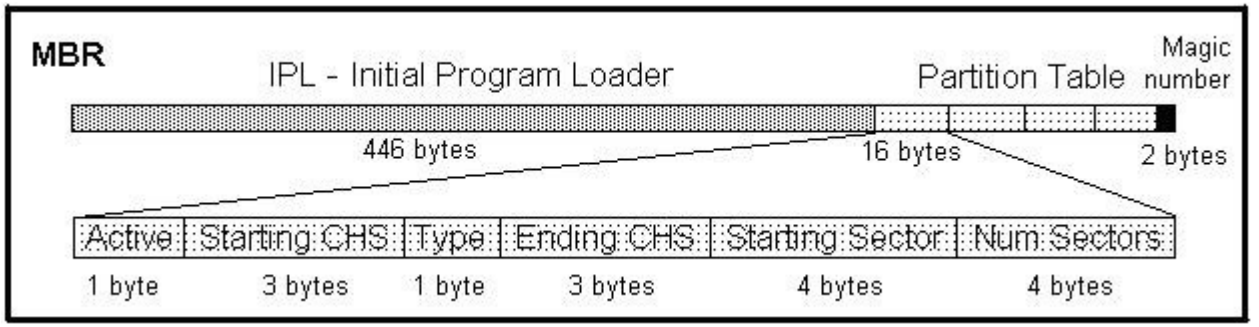

Устройство шинного интерфейса
15
0
CS
SS
DS
ES
Сумматор адресов
IP
Буфер адреса/ состояния
A19 - A16
ST6 - ST3
Буфер адреса/ данных
AD15 - AD0
Операционное устройство
15
8
7
0
AX
CX
DX
BX
AH
AL
CH
CL
DH
DL
BH
BL
SP
BP
SI
DI
АЛУ
15
0
F
7
0
1
Очередь команд
IQ
6
Устройство управления
BHE/ST7
ALE(QS0)
DEN(ST0)
DT/R(ST1)
M/IO(ST2)
WR(LOCK)
RD
MN/MX
CLK
TEST
HOLD (RQ/ST0)
NMI
INTA(QS1)
RESET
READY
HLDA(RQ/GT1)
INTR


Clock cycle
0
1
2
3
4
5
6
7
8
Waiting instructions
Pipeline
Stage 1: Fetch
Stage 2: Decode
Stage 3: Execute
Stage 4: Write-back
Completed instructions

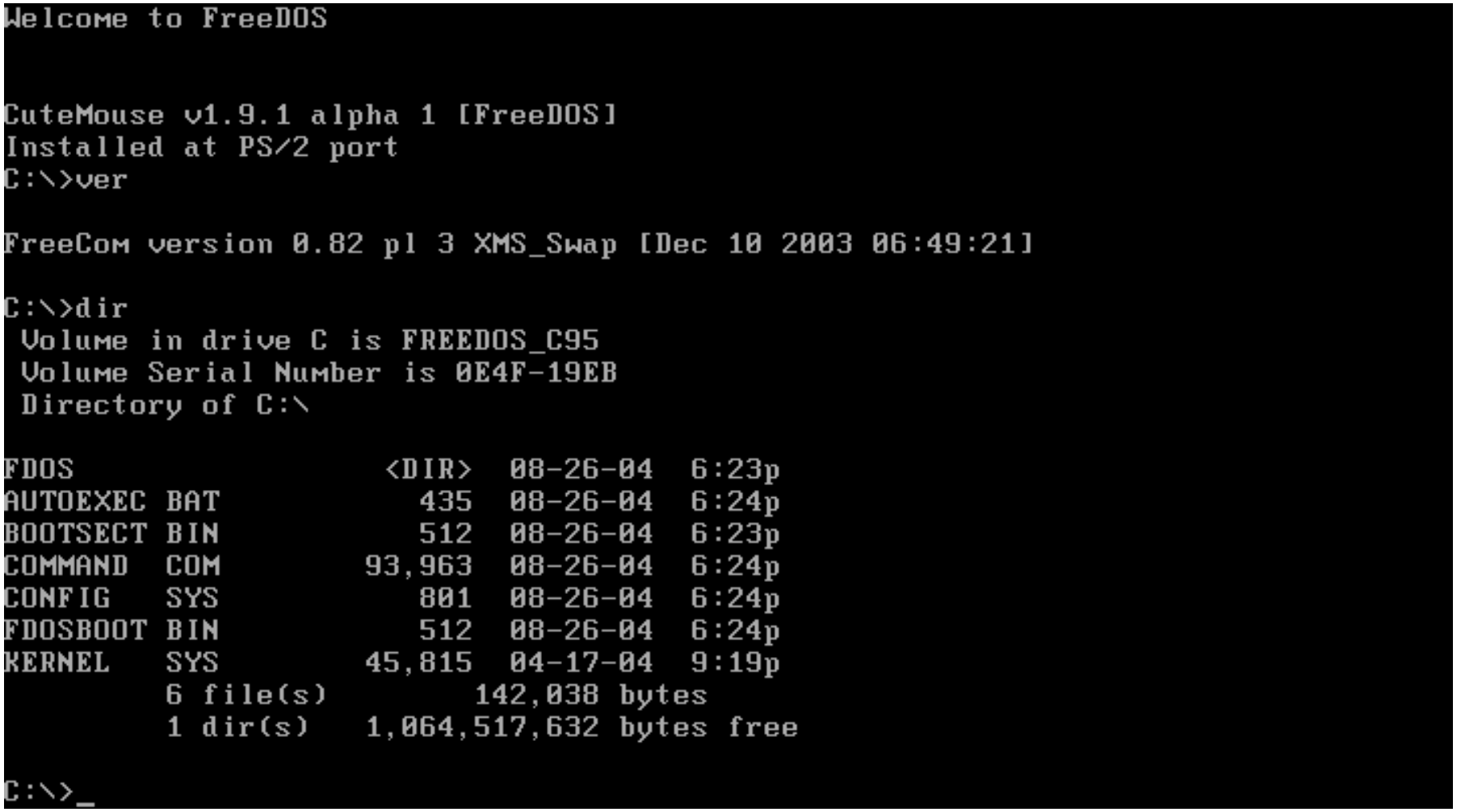
We have a hardware, what is next?

We have a hardware, what is next?

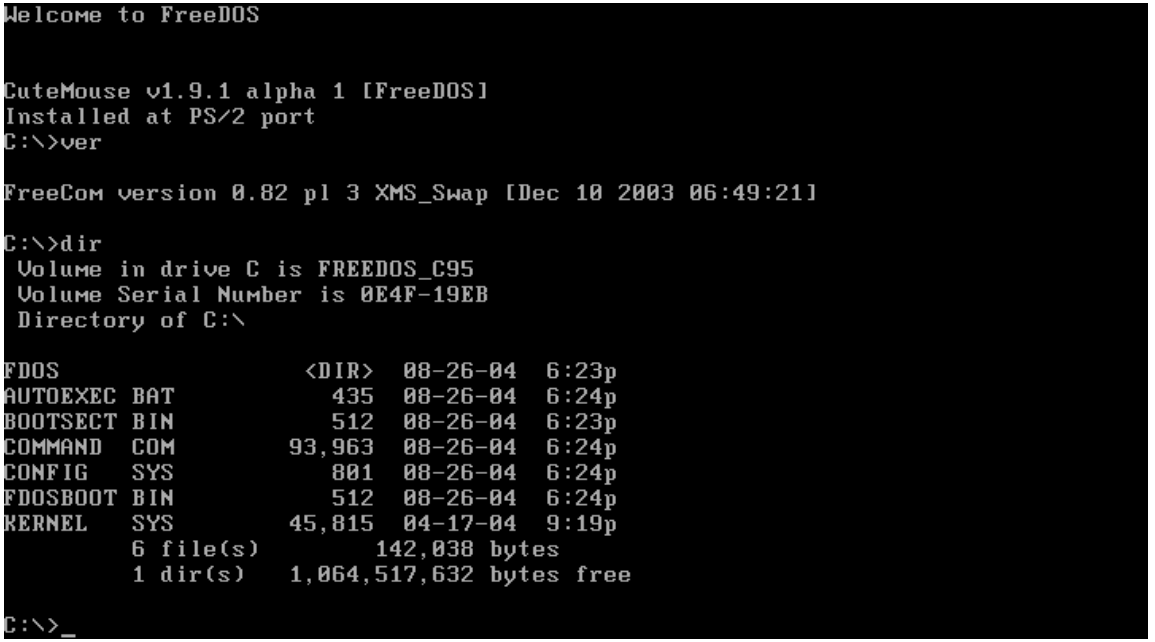




Disk Operating System



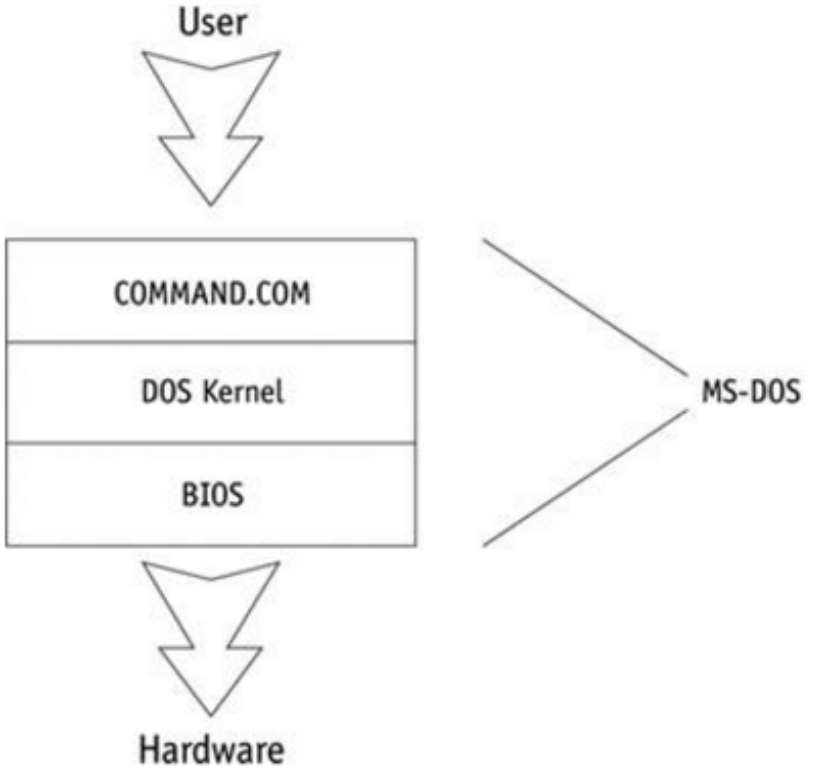
Disk Operating System



- MS DOS memory limit (640KB ought to be enough for anybody)
- No memory protection, first-fit memory allocation scheme
- Single task, but TSR (Terminate and Stay Resident)
- Single user

Disk Operating System

```
Welcome to FreeDOS


CuteMouse v1.9.1 alpha 1 [FreeDOS]
Installed at PS/2 port
C:\>ver

FreeCom version 0.82 pl 3 XMS_Swap [Dec 10 2003 06:49:21]

C:\>dir
 Volume in drive C is FREEDOS_C95
 Volume Serial Number is 0E4F-19EB
 Directory of C:\

FDOS                <DIR>  08-26-04  6:23p
AUTOEXEC BAT          435  08-26-04  6:24p
BOOTSECT BIN          512  08-26-04  6:23p
COMMAND  COM       93,963  08-26-04  6:24p
CONFIG   SYS          801  08-26-04  6:24p
FDOSBOOT BIN          512  08-26-04  6:24p
KERNEL   SYS       45,815  04-17-04  9:19p
        6 file(s)        142,038 bytes
        1 dir(s)   1,064,517,632 bytes free

C:\>_
```



BIOS

- **BIOS (Basic Input/Output System)**
  - Direct interface with I/O devices
  - Contains device drivers
    - Controls data flow to and from each device (except disk drives)
  - Receives I/O operation status information
    - Passes to processor
  - Handles small differences among I/O units
    - No need to write device driver for manufacturer printer

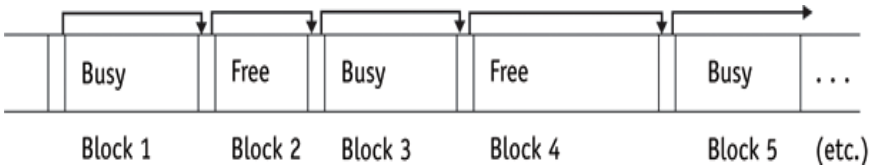
DOS kernel

- **DOS kernel**
  - Contains routines to interface with disk drives
  - Read into memory
    - Initialization time from MSDOS.SYS file
    - Resides in boot disk
  - Microsoft proprietary program
  - Accessed by application programs
  - Provides hardware-independent services
    - System functions
    - Memory management, file and record management
  - Provides transparency
    - Compensates for manufacturer variations
  - Manages file storage and retrieval
  - Dynamically allocates and deallocates secondary storage as needed

Command processor

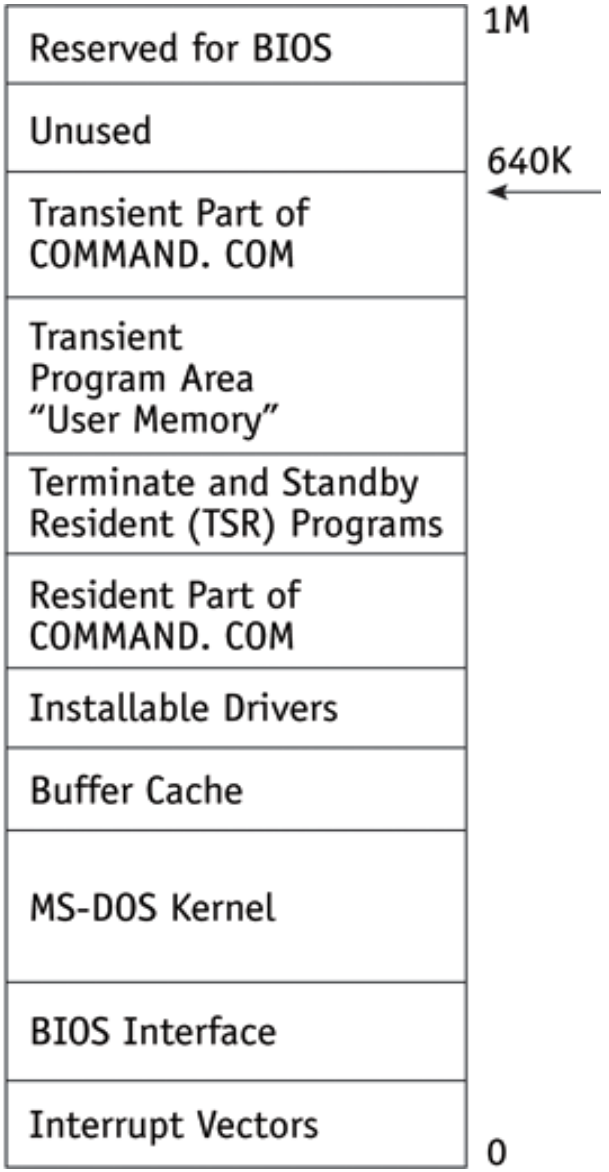
- **Command processor (shell)**
  - Sends prompts to user
  - Accepts typed commands
  - Executes commands
    - From **system prompt**
    - Issues appropriate responses
  - Resides in COMMAND.COM file
    - Stored in two different main memory sections
  - Appears on public directory
  - Weakness: not interpretive

- Main memory structure
  - **ROM**
    - Very small in size
    - Contains program
    - Contains section of BIOS with startup process (bootstrapping)
    - Initializes computer
    - Retrieves resident code and loads into RAM
  - **RAM**
    - Part of main memory
    - Where programs are loaded and executed



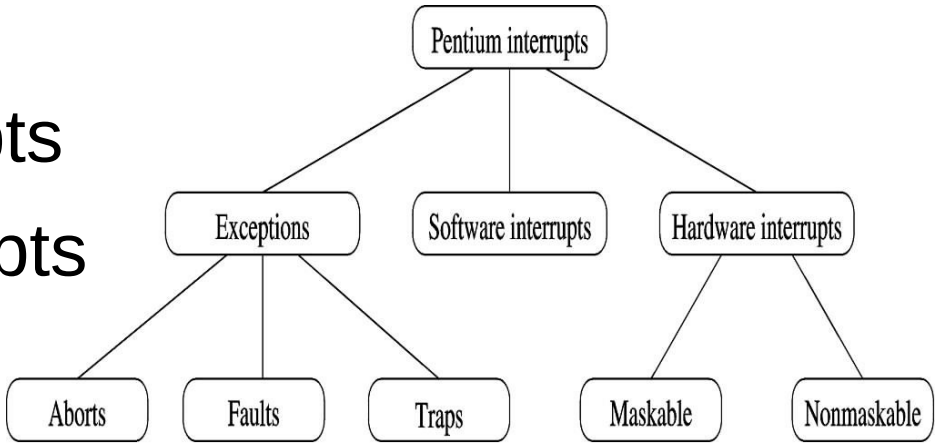
(figure 14.4)

The linked list of memory blocks.

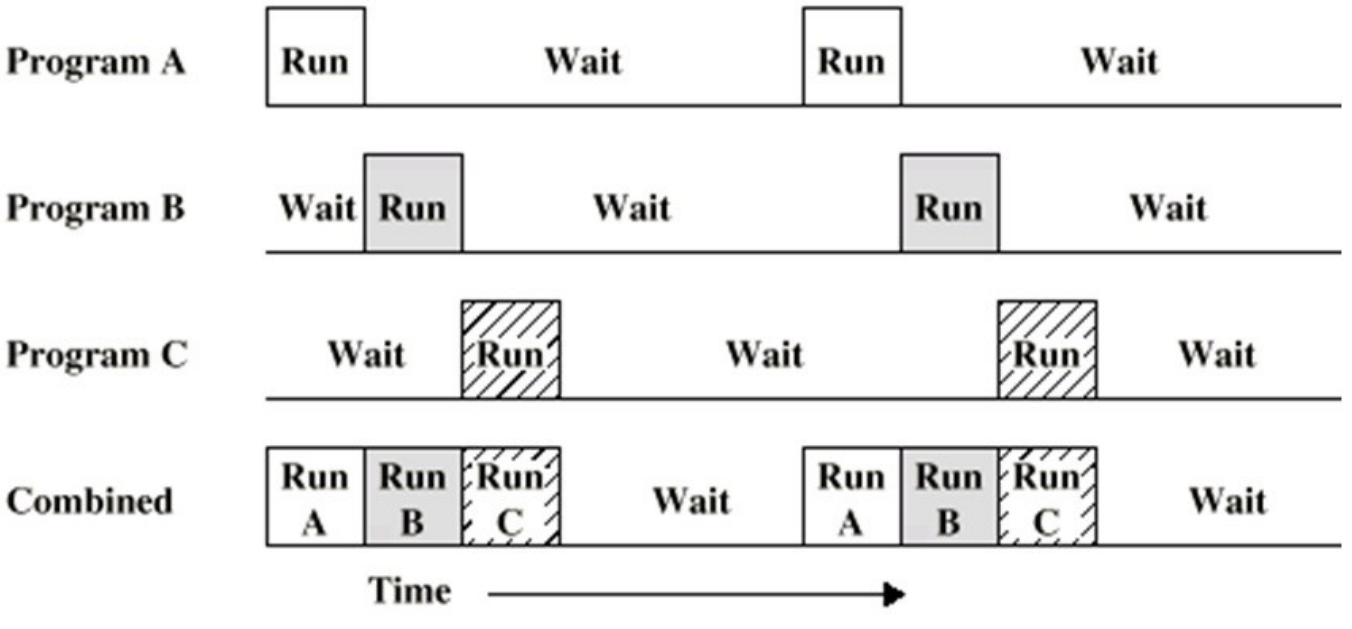


Single user, single task

- Interrupts:
  - Internal hardware interrupts
  - External hardware interrupts
  - Software interrupts



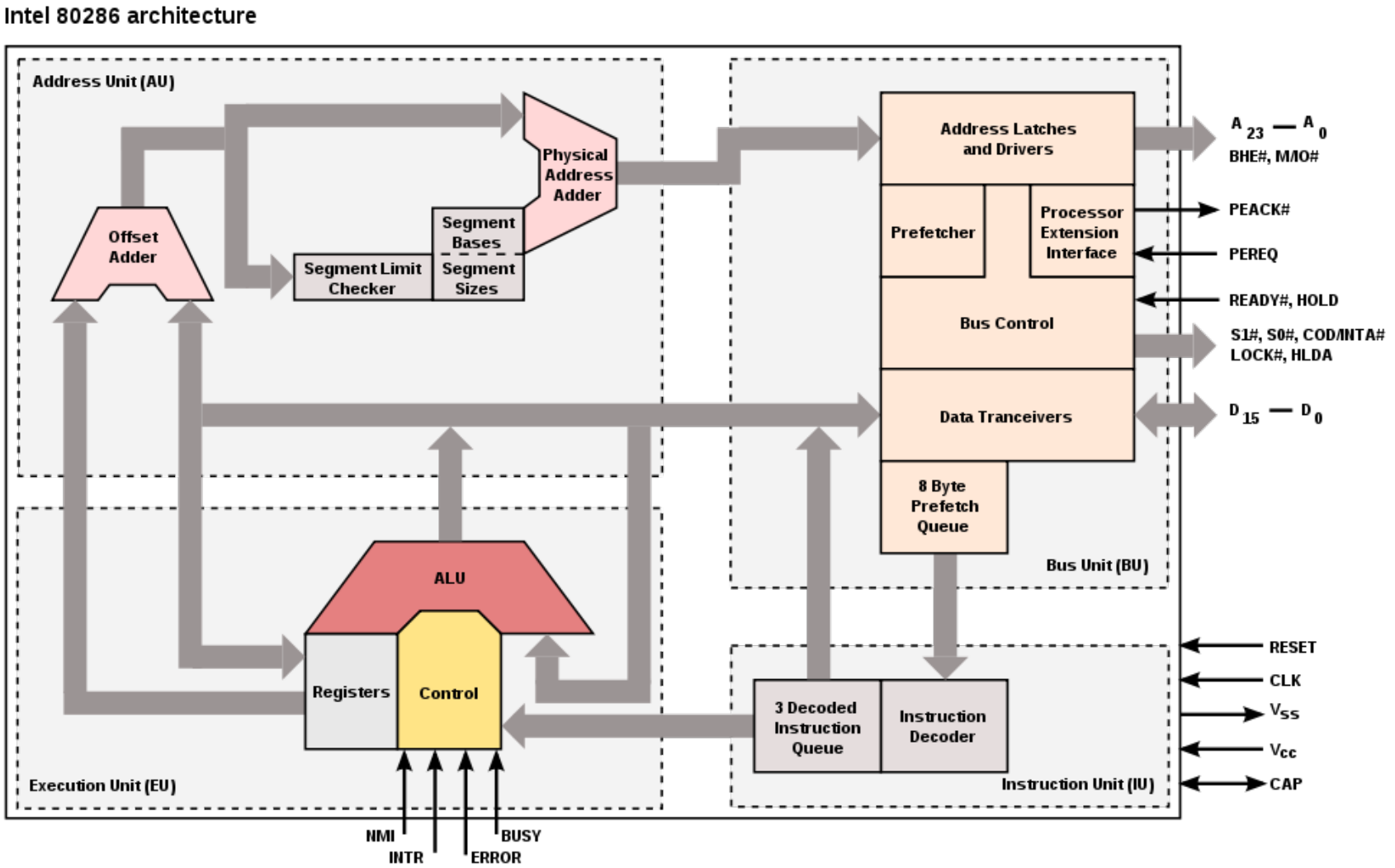
Multitasking:



- Single user, signle task
- Unsafe, unsecure
- A lot of code duplication
- etc…

Required to add hardware features!

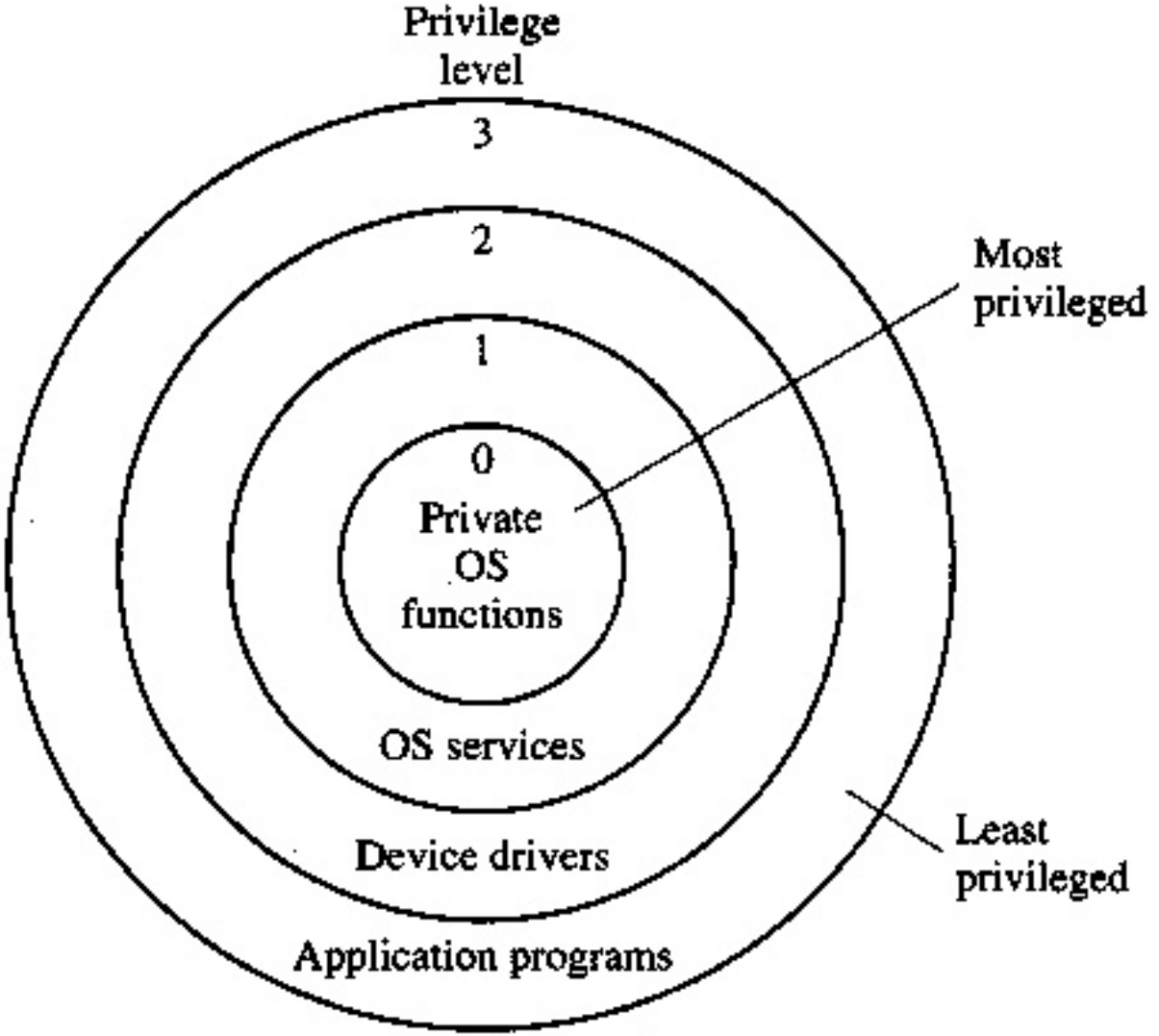
- 80286: protected mode, virtual memory, 20MHz



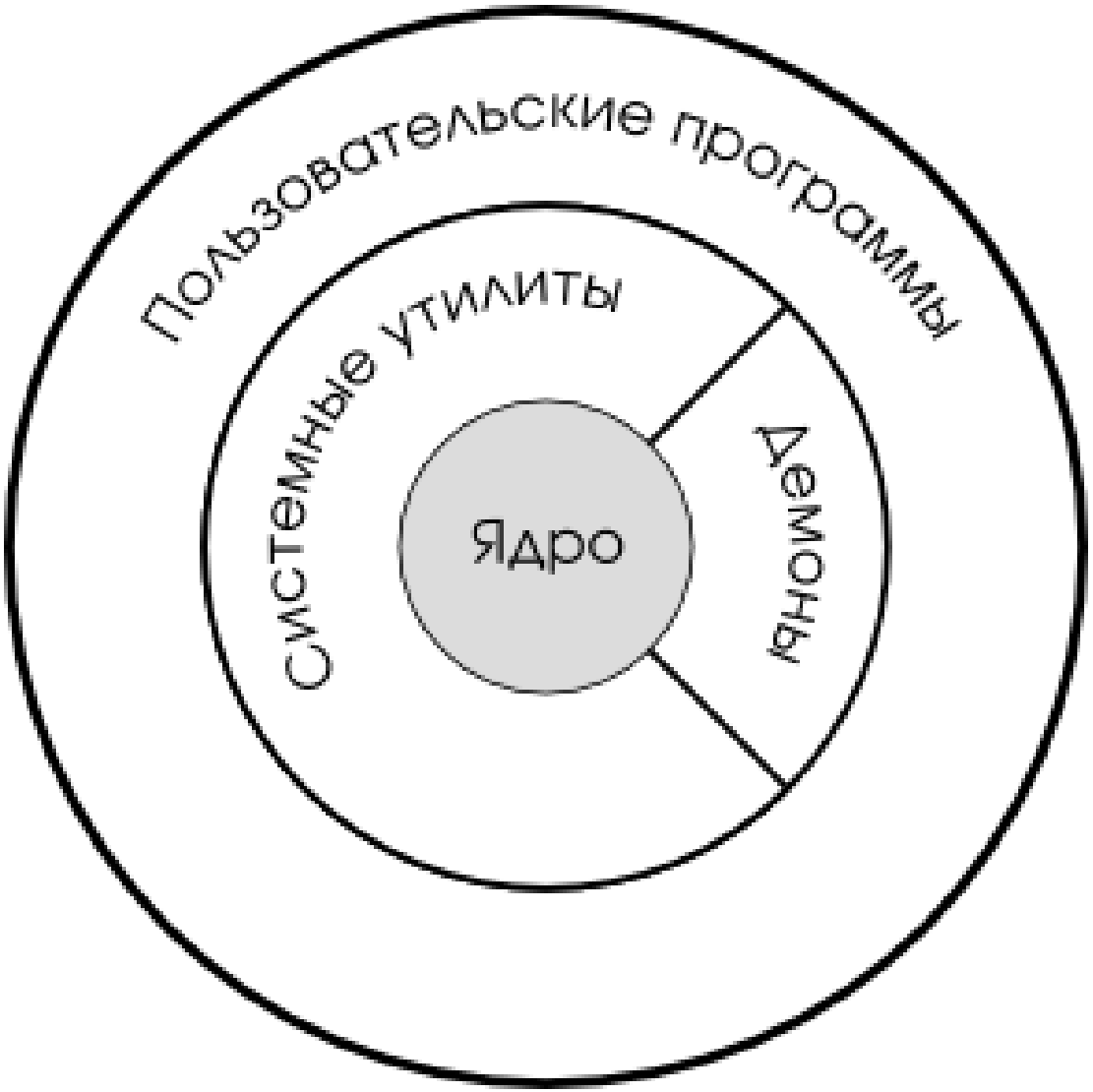
- protected mode, virtual memory

- protected mode, virtual memory

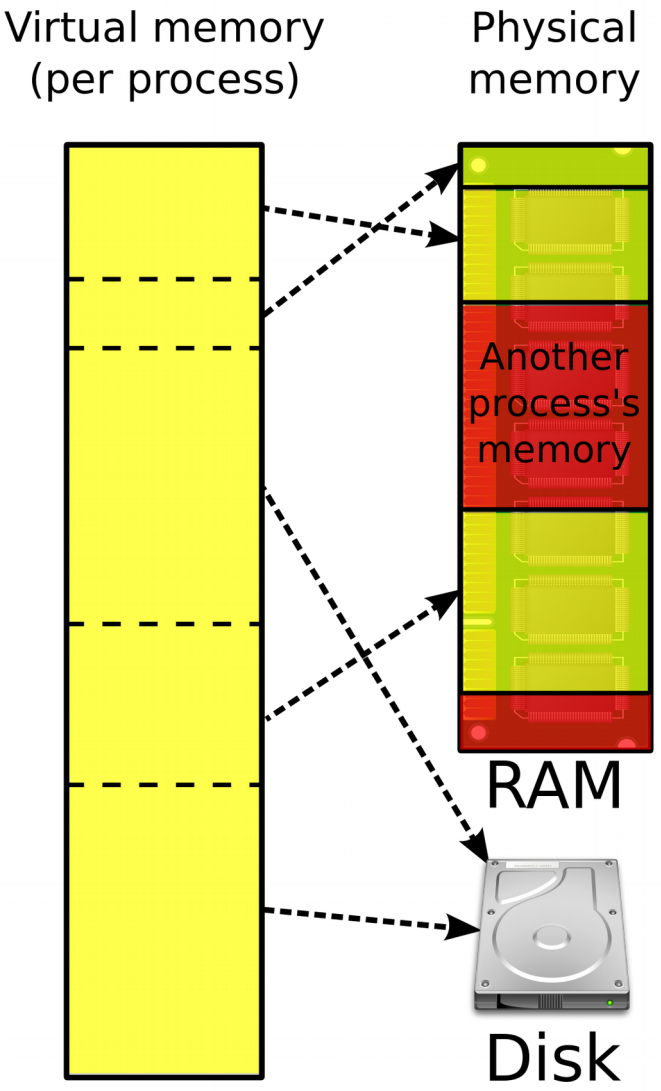
Hardware level:

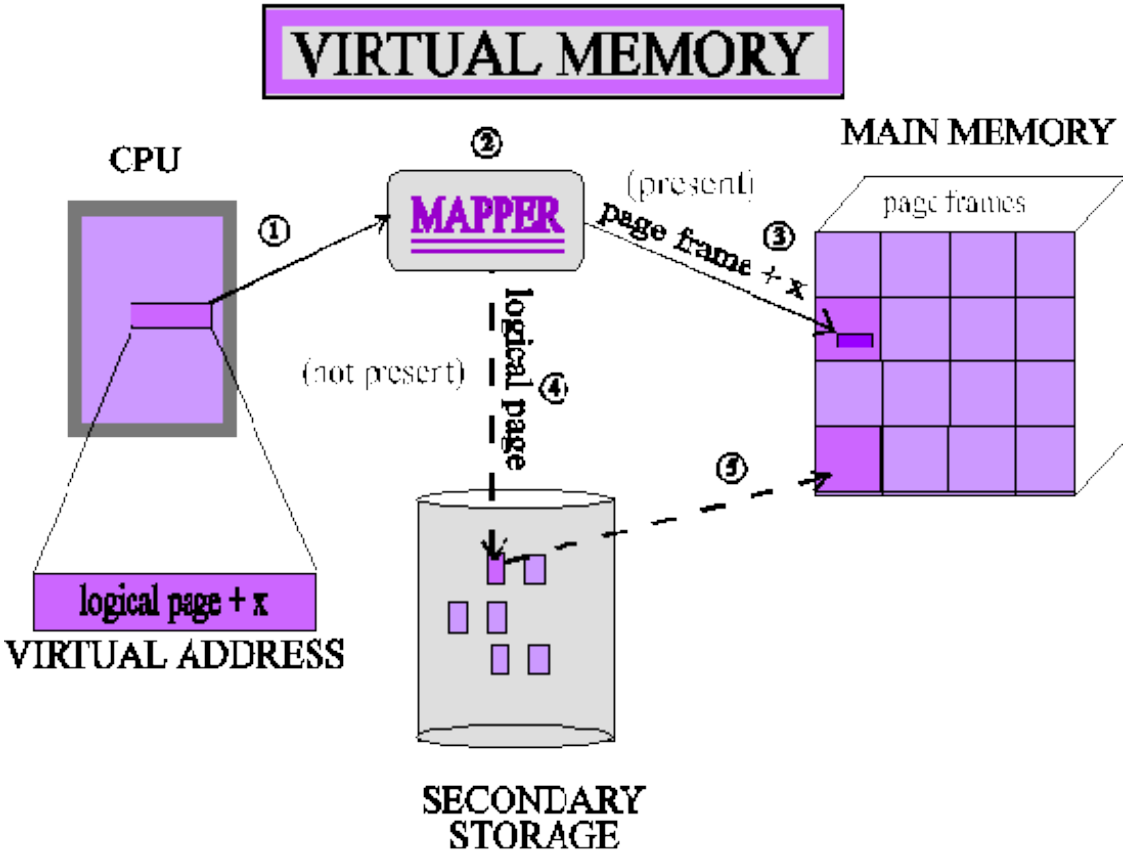


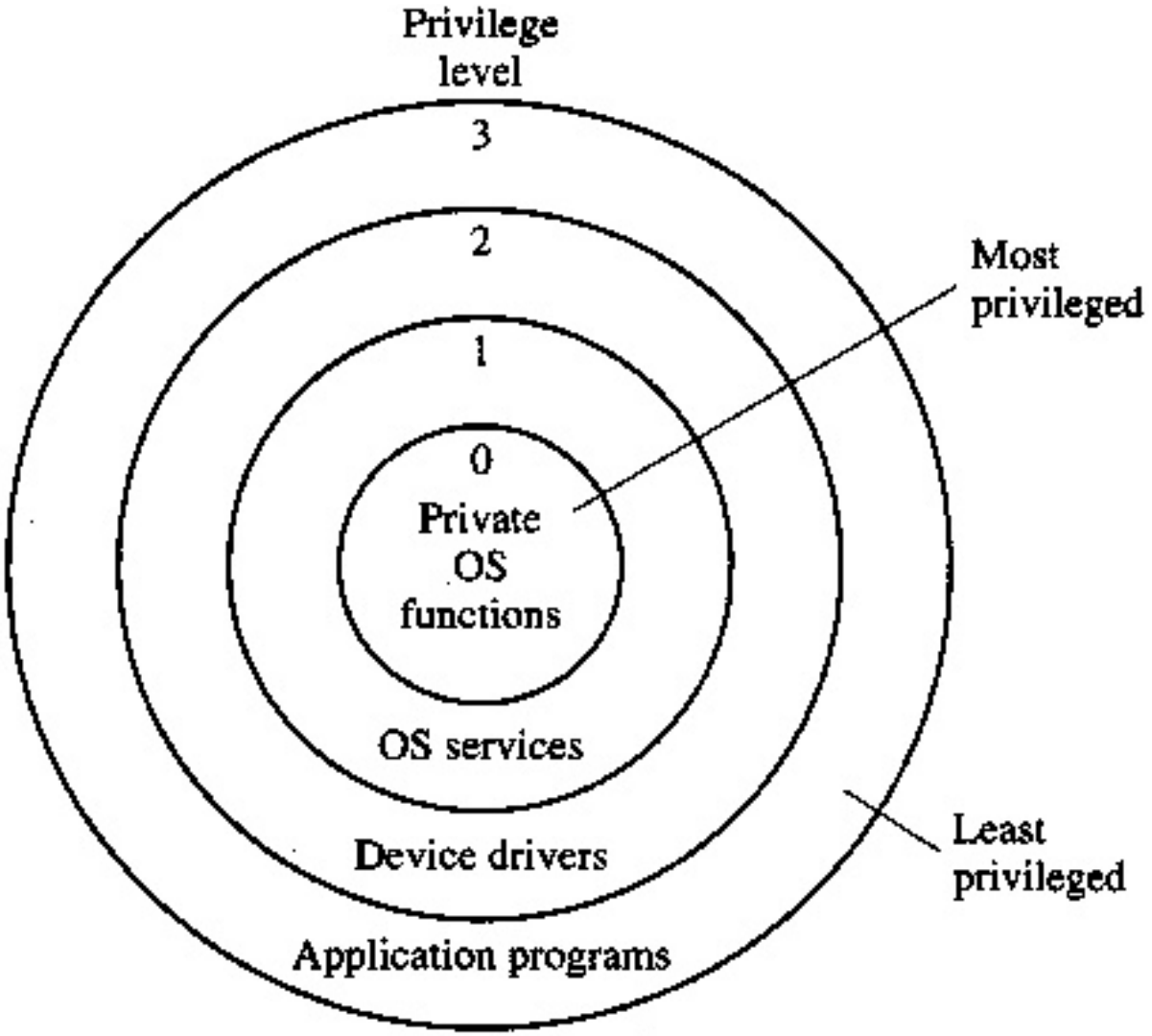
Software level:

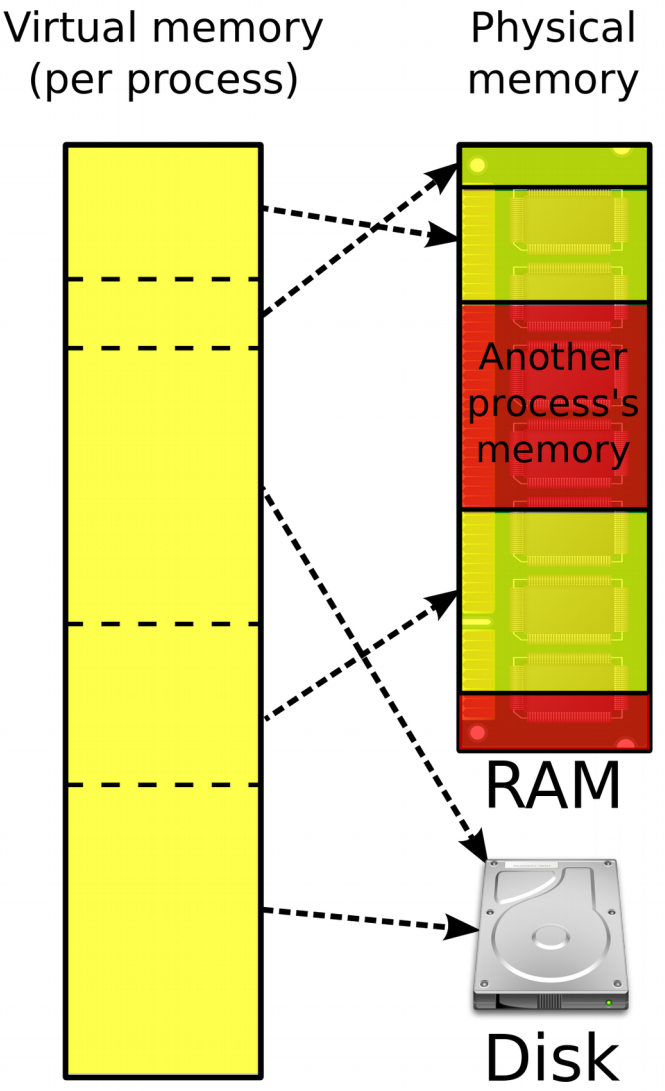


- protected mode, virtual memory



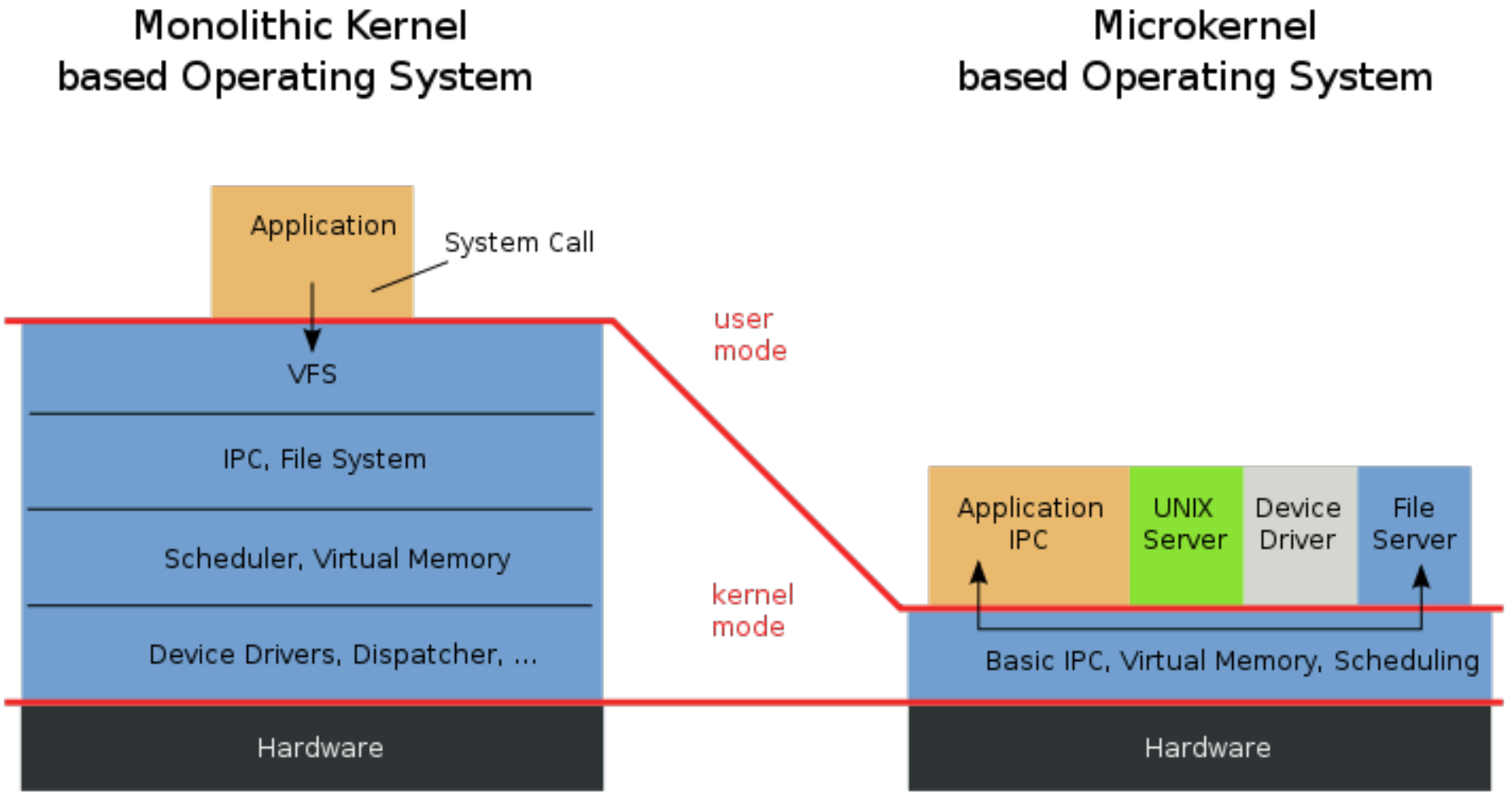






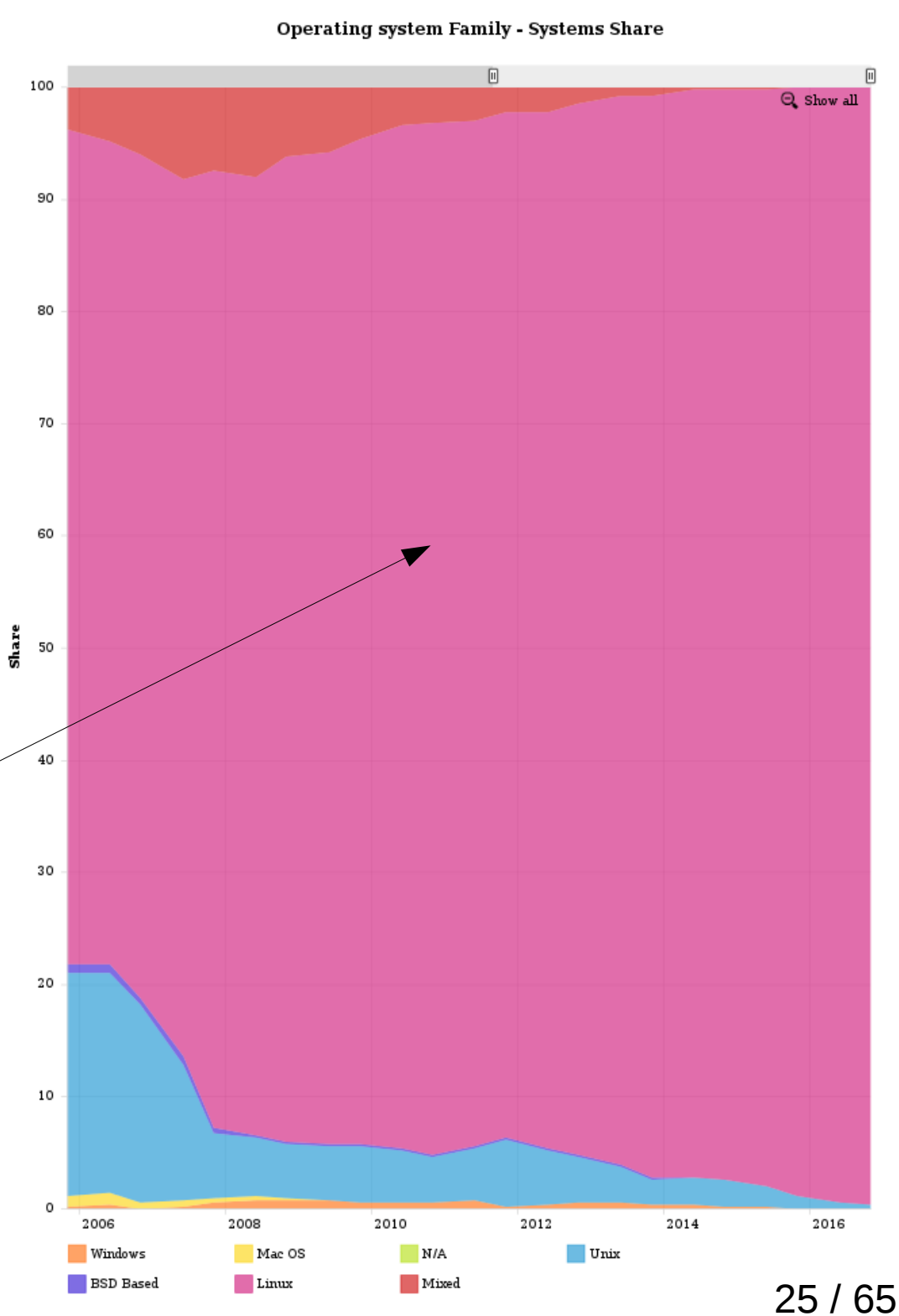
Again: we have a hardware, what is next?

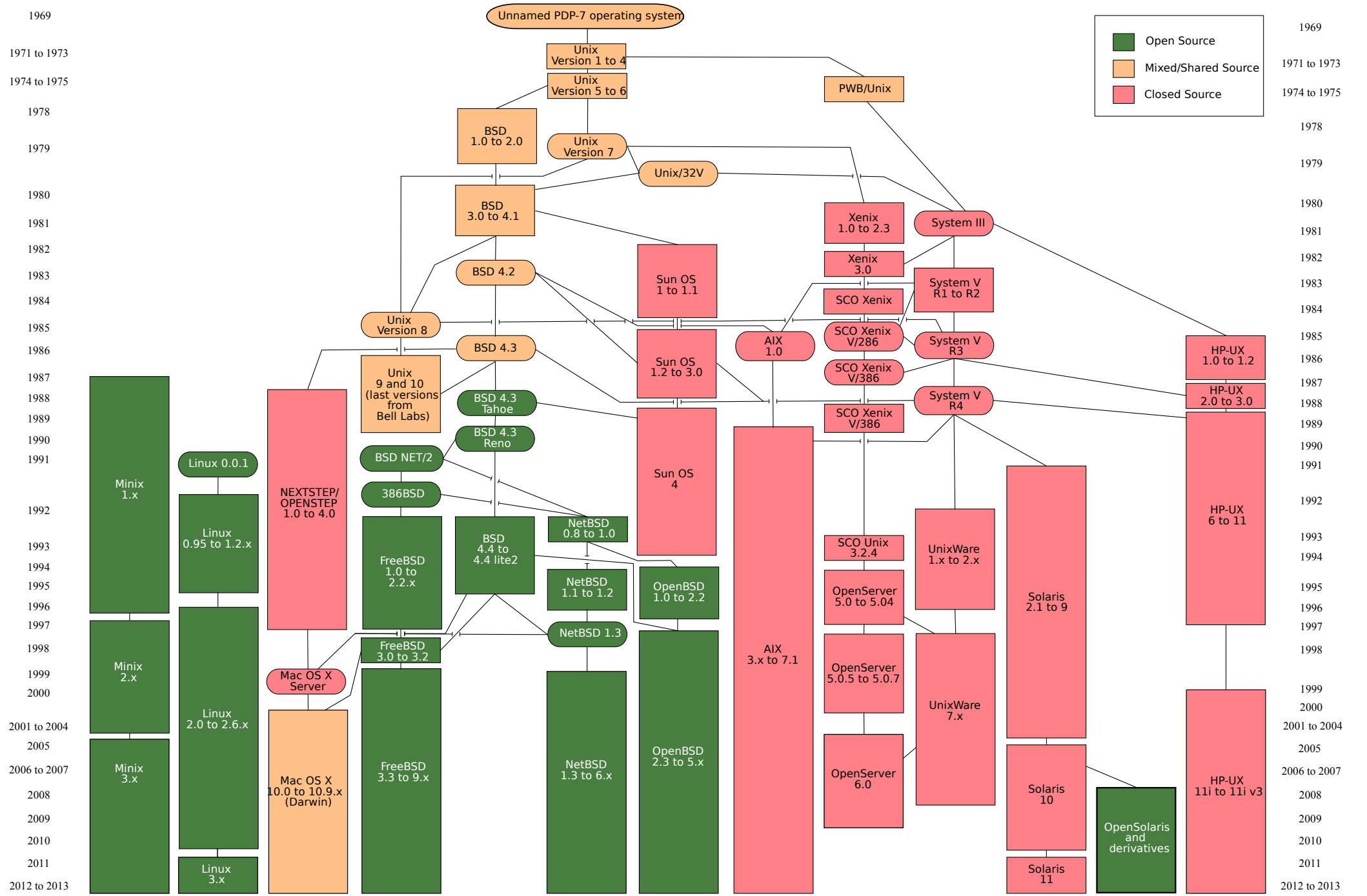
Monolithic vs Microkernel



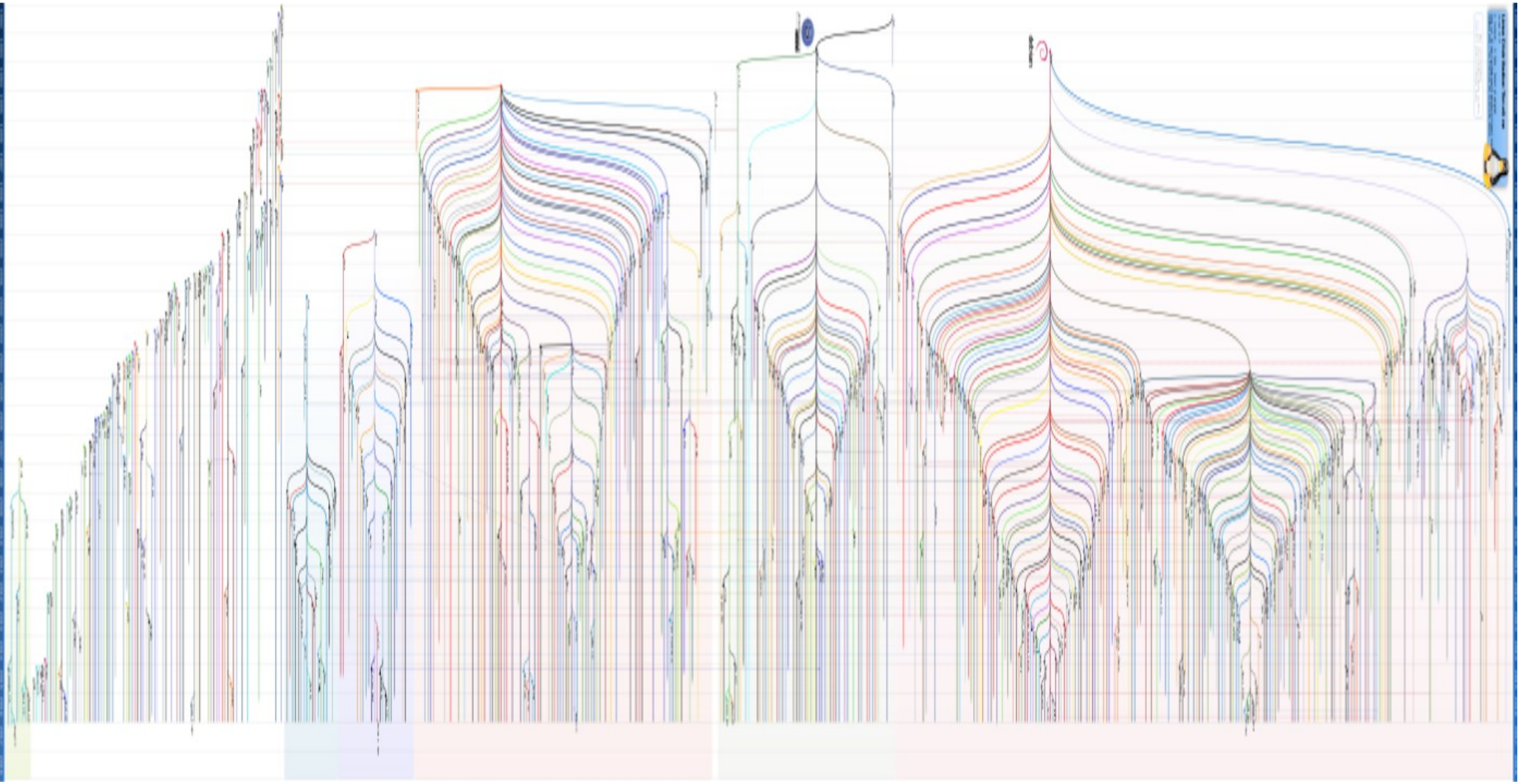
Tenenbaum-Torvalds debate

Linux (in HPC market share)




Open Source
Mixed/Shared Source
Closed Source
1969
1971 to 1973
1974 to 1975
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001 to 2004
2005
2006 to 2007
2008
2009
2010
2011
2012 to 2013
Unnamed PDP-7 operating system
Unix Version 1 to 4
Unix Version 5 to 6
Unix Version 7
PWB/Unix
BSD 1.0 to 2.0
Unix/32V
BSD 3.0 to 4.1
Xenix 1.0 to 2.3
Xenix 3.0
System III
BSD 4.2
Sun OS 1 to 1.1
SCO Xenix
System V R1 to R2
Unix Version 8
BSD 4.3
Sun OS 1.2 to 3.0
AIX 1.0
SCO Xenix V/286
SCO Xenix V/386
System V R3
HP-UX 1.0 to 1.2
Unix 9 and 10 (last versions from Bell Labs)
BSD 4.3 Tahoe
BSD 4.3 Reno
System V R4
HP-UX 2.0 to 3.0
BSD NET/2
386BSD
Minix 1.x
Linux 0.0.1
NEXTSTEP/ OPENSTEP 1.0 to 4.0
Linux 0.95 to 1.2.x
FreeBSD 1.0 to 2.2.x
BSD 4.4 to 4.4 lite2
NetBSD 0.8 to 1.0
NetBSD 1.1 to 1.2
OpenBSD 1.0 to 2.2
NetBSD 1.3
FreeBSD 3.0 to 3.2
Sun OS 4
AIX 3.x to 7.1
SCO Unix 3.2.4
OpenServer 5.0 to 5.04
OpenServer 5.0.5 to 5.0.7
OpenServer 6.0
UnixWare 1.x to 2.x
UnixWare 7.x
Solaris 2.1 to 9
Solaris 10
Solaris 11
HP-UX 6 to 11
HP-UX 11i to 11i v3
Minix 2.x
Minix 3.x
Linux 2.0 to 2.6.x
Linux 3.x
Mac OS X Server
Mac OS X 10.0 to 10.9.x (Darwin)
FreeBSD 3.3 to 9.x
NetBSD 1.3 to 6.x
OpenBSD 2.3 to 5.x
OpenSolaris and derivatives

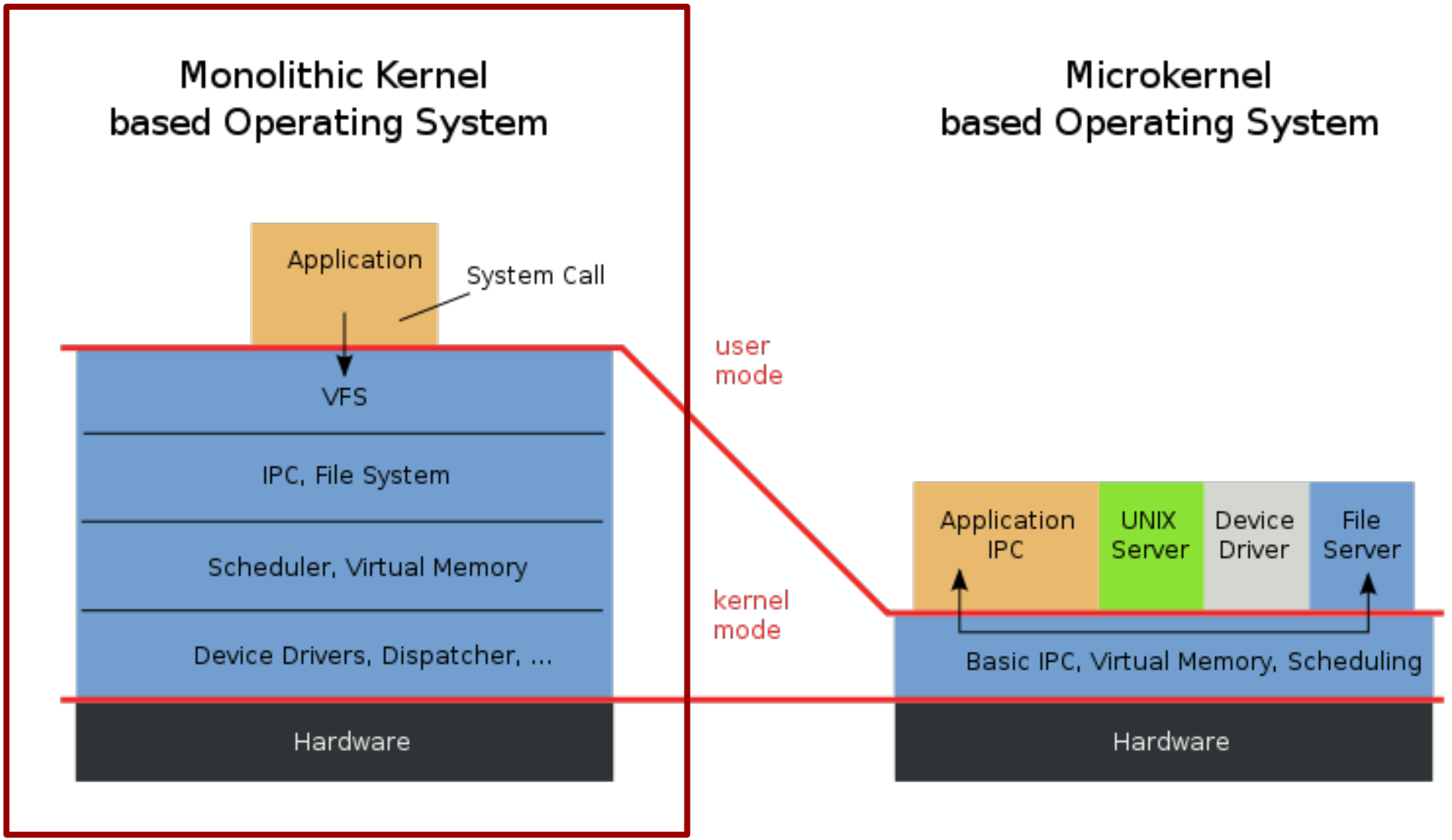
Linux distributions

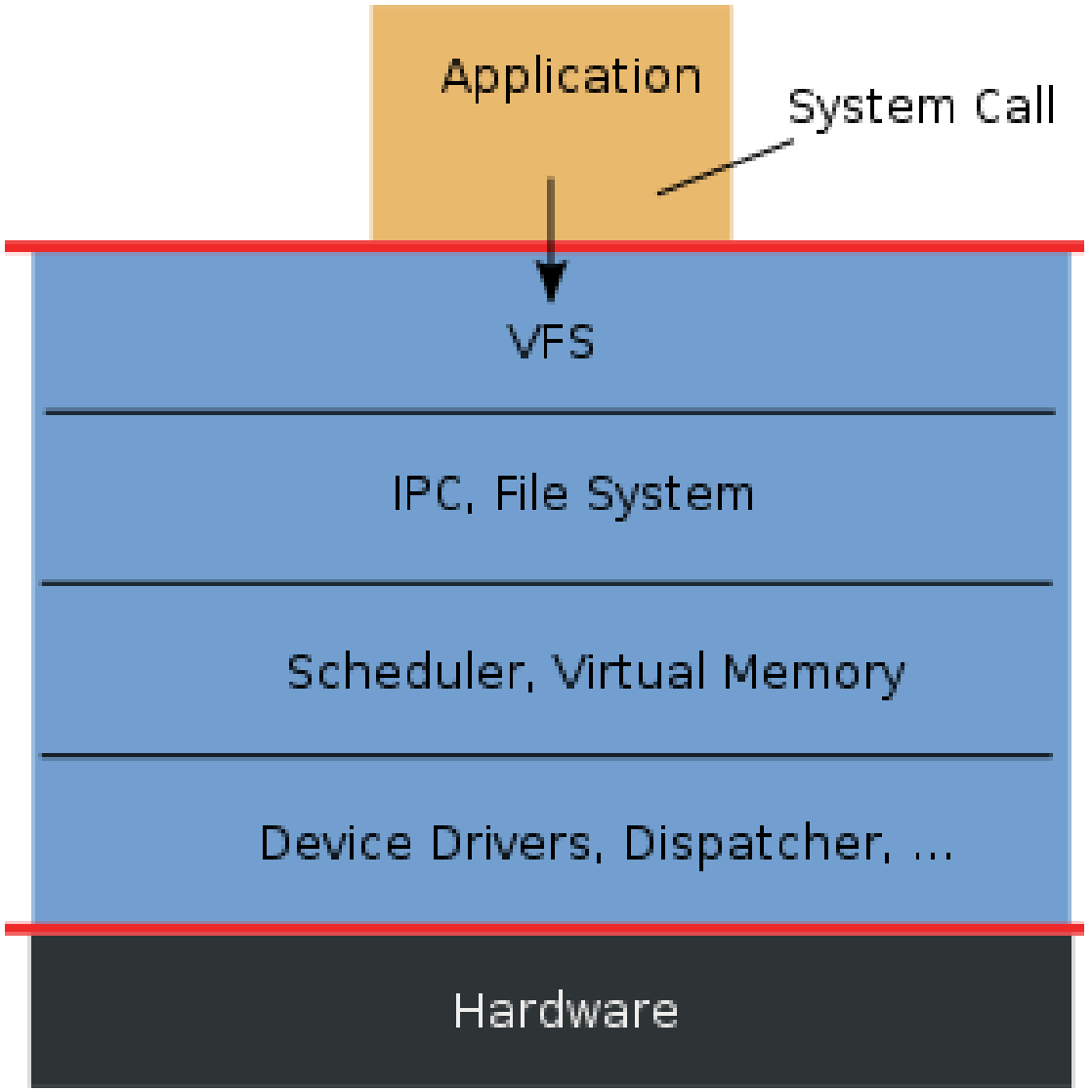


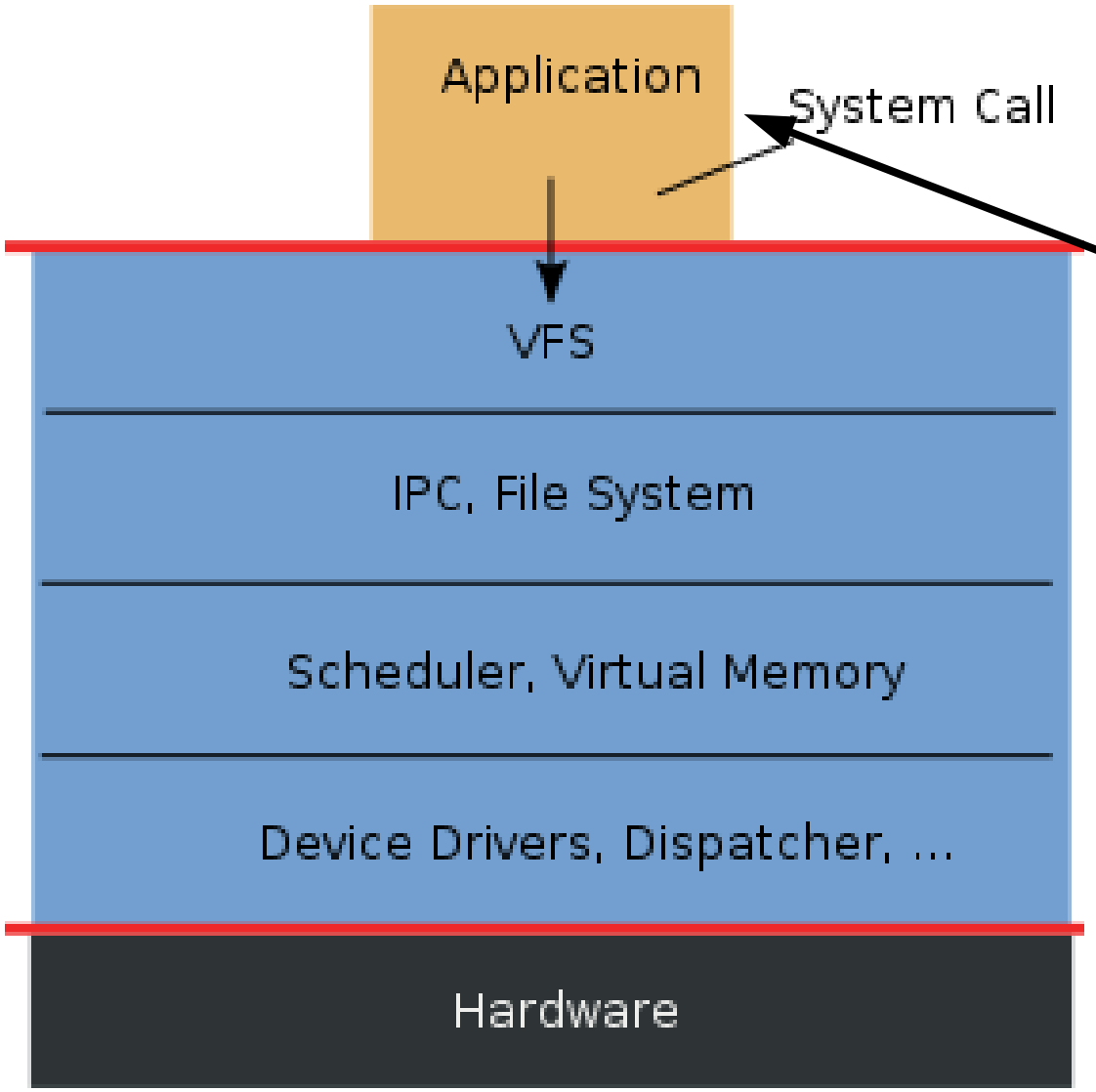
Must read

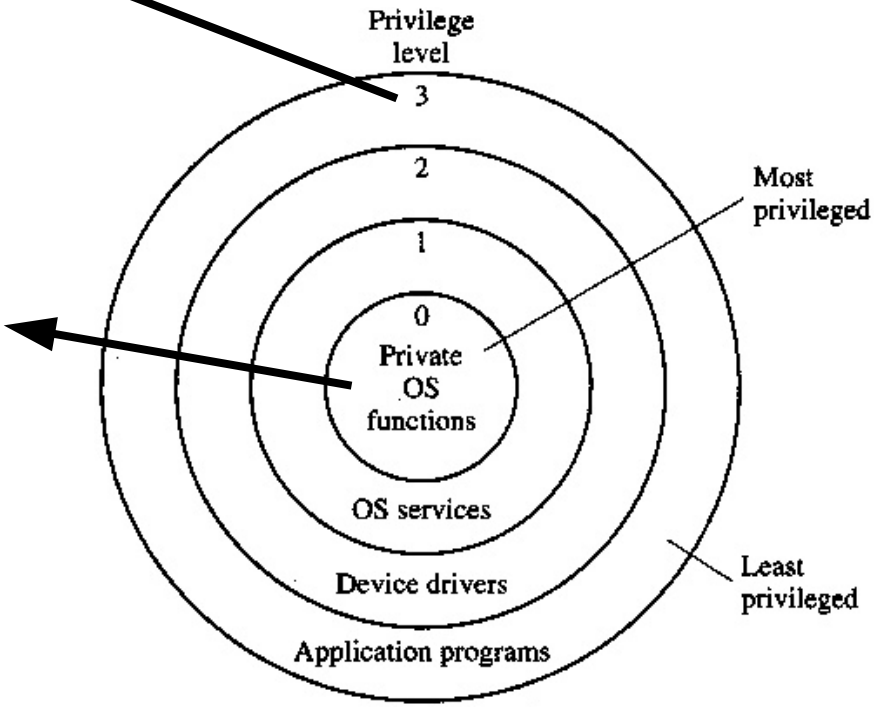
- Tenenbaum-Torvalds debate
- GNU C Library debates about Versioned Interfaces

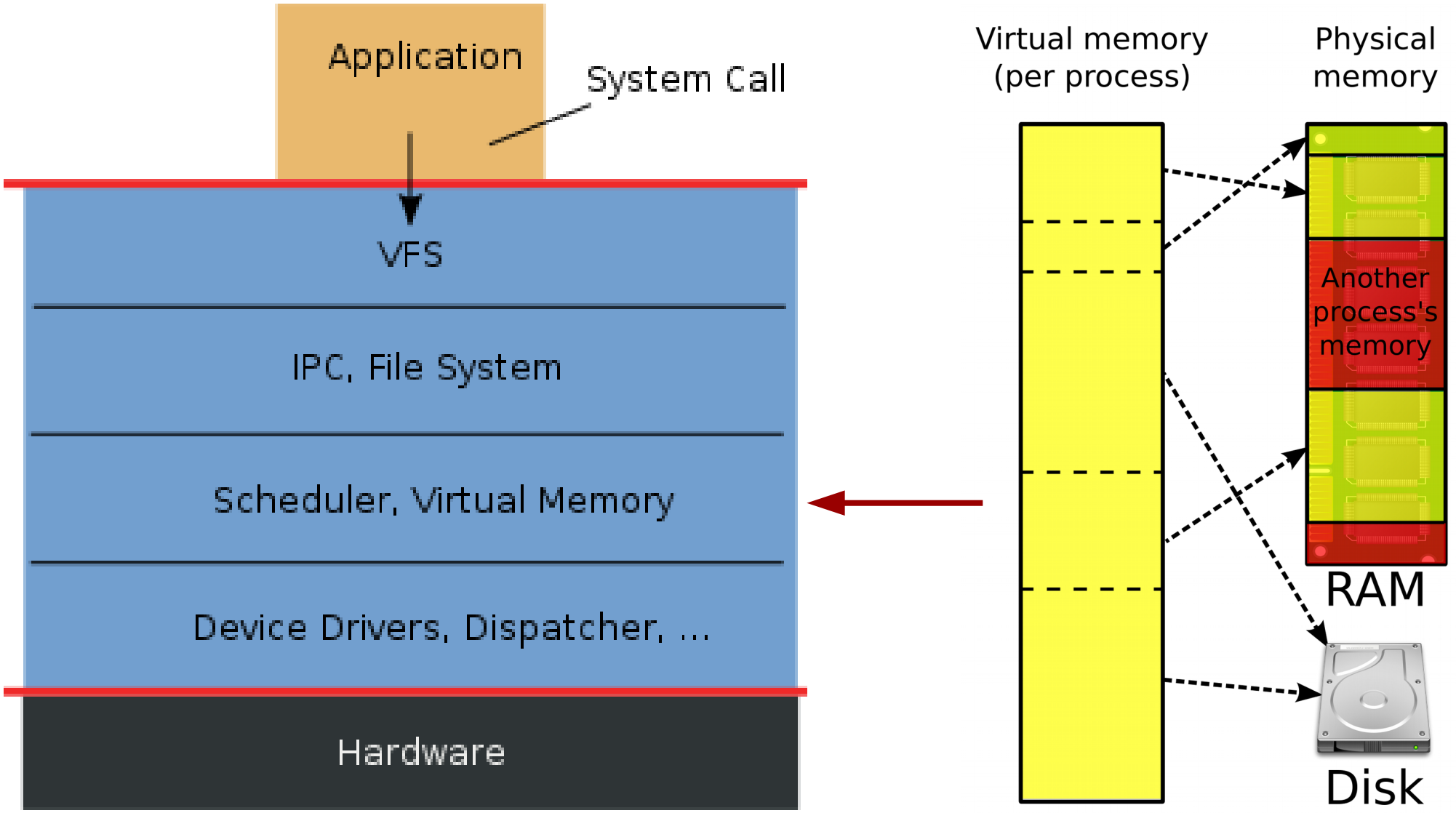
Monolithic vs Microkernel



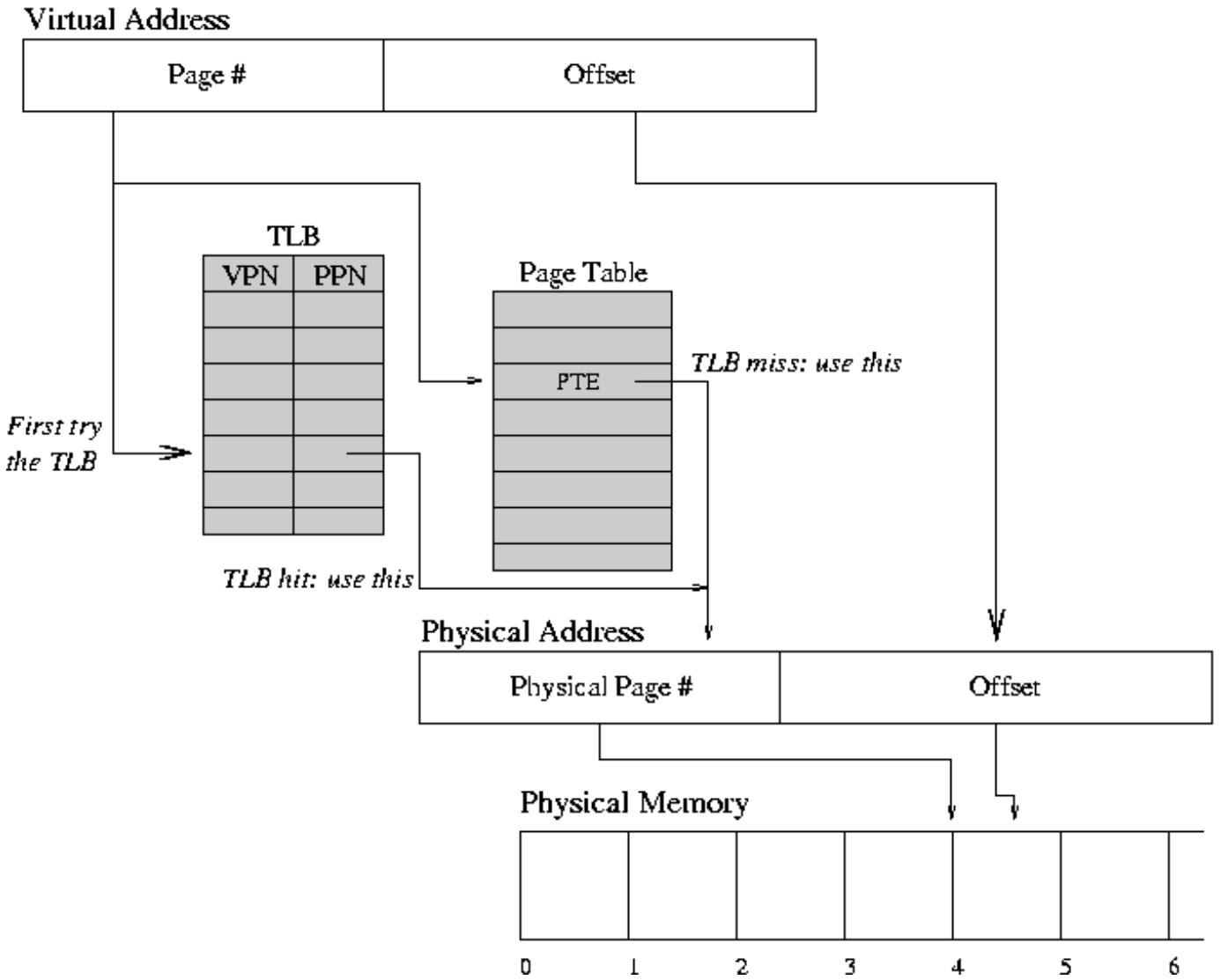

Application
System Call
VFS
IPC, File System
Scheduler, Virtual Memory
Device Drivers, Dispatcher, ...
Hardware


Application
System Call
VFS
IPC, File System
Scheduler, Virtual Memory
Device Drivers, Dispatcher, ...
Hardware

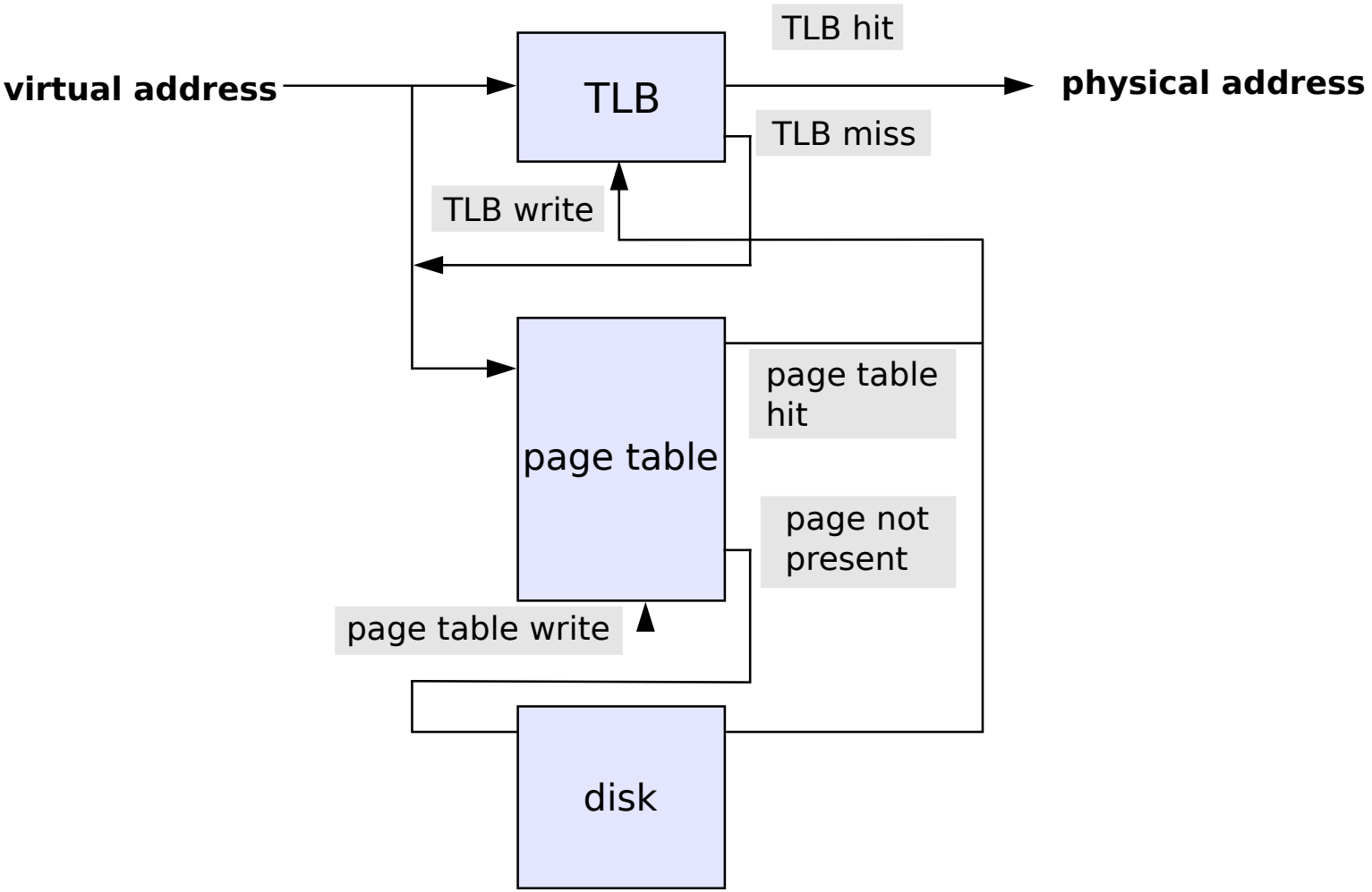

Privilege level
3
2
1
0
Private OS functions
OS services
Device drivers
Application programs
Most privileged
Least privileged


Application
System Call
VFS
IPC, File System
Scheduler, Virtual Memory
Device Drivers, Dispatcher, ...
Hardware
Virtual memory
(per process)
Physical
memory
Another
process's
memory
RAM
Disk

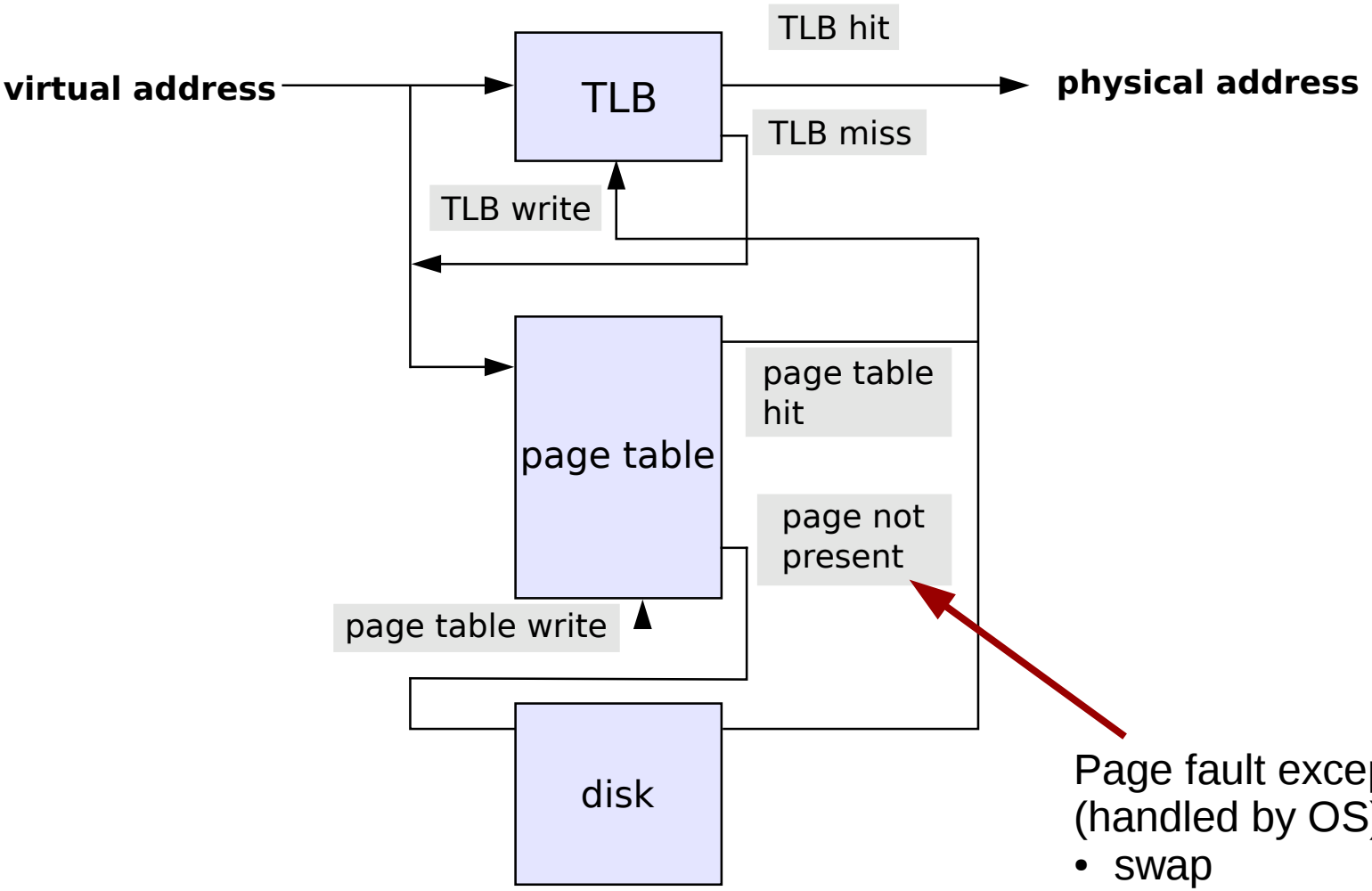
Virtual memory



Virtual memory



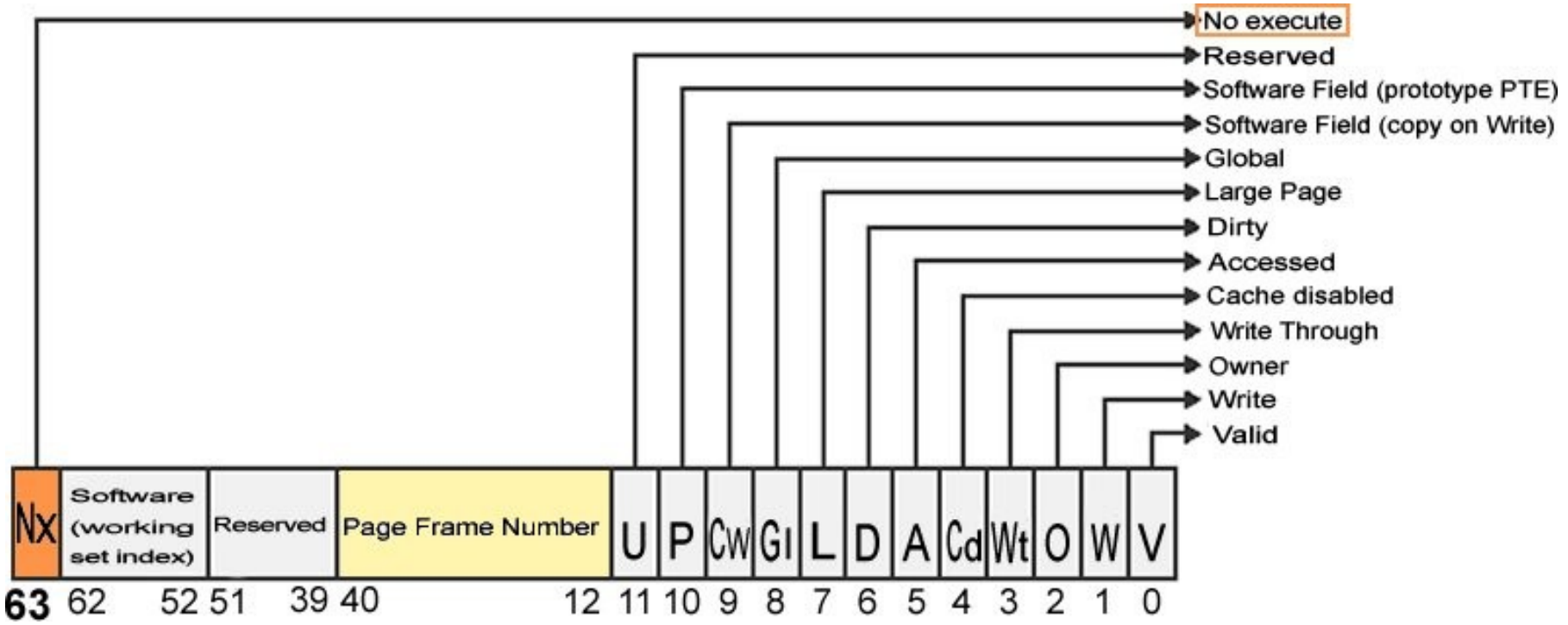
Virtual memory

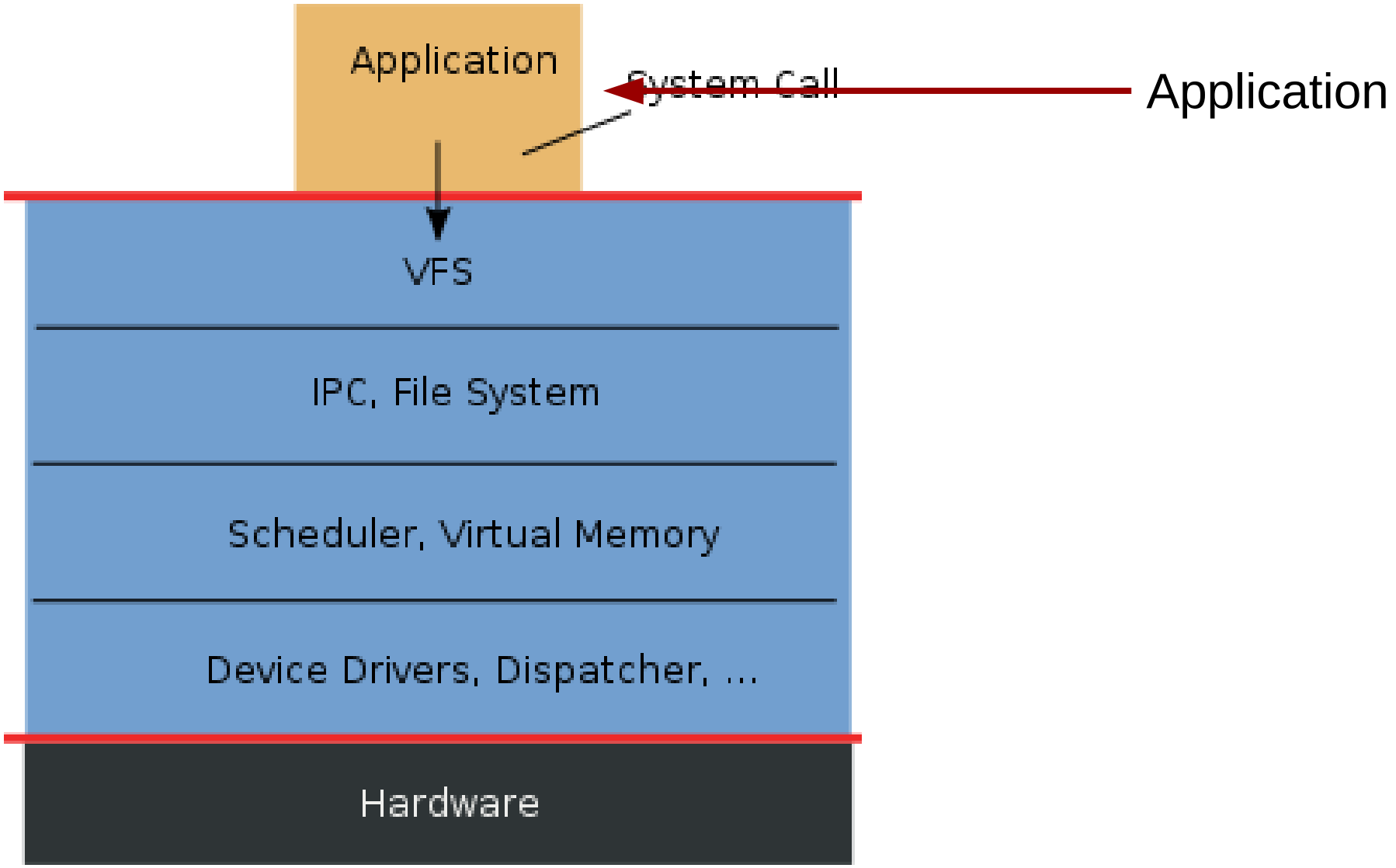


Page fault exception (handled by OS):

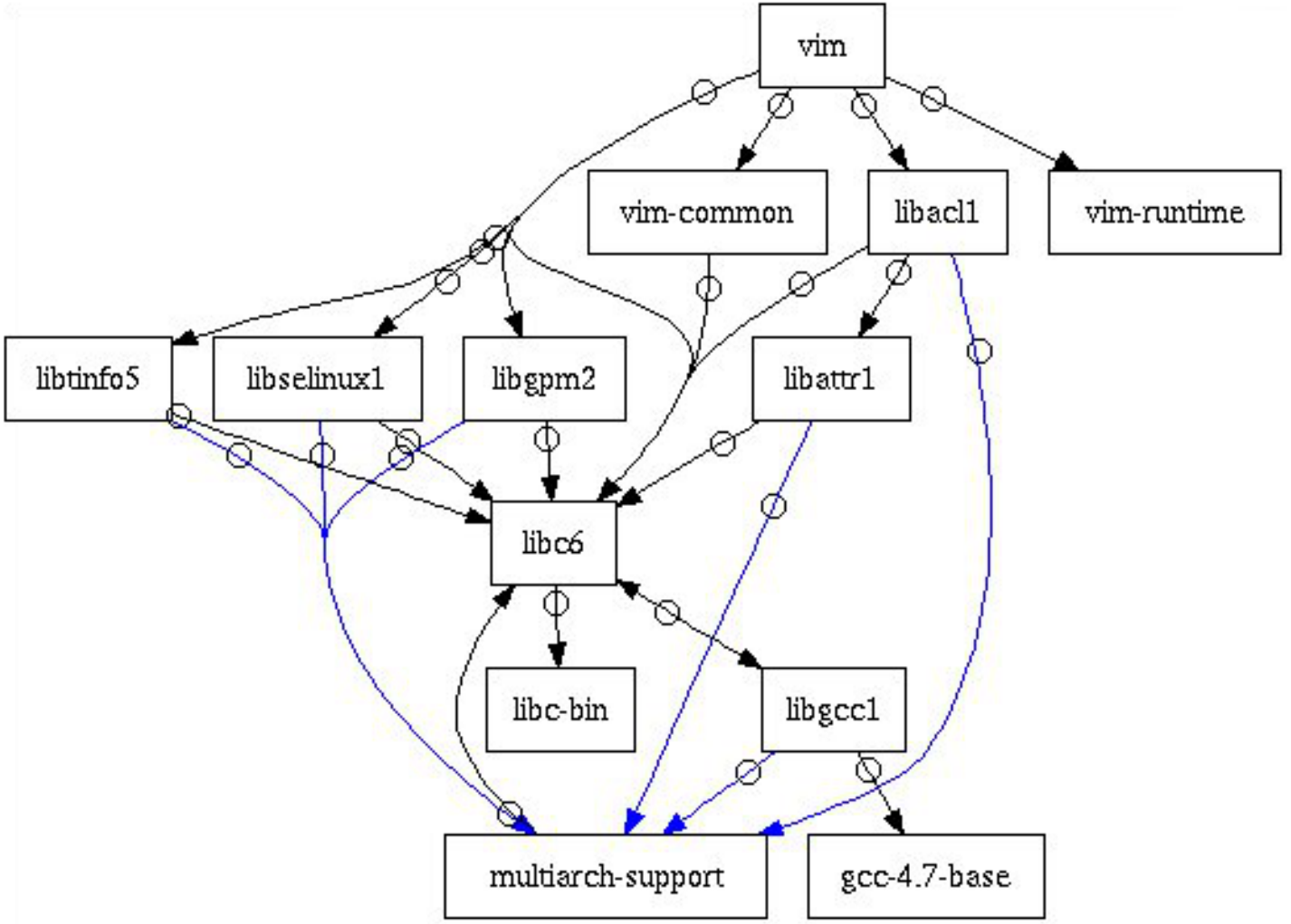
- swap
- zRAM
- etc

Virtual memory

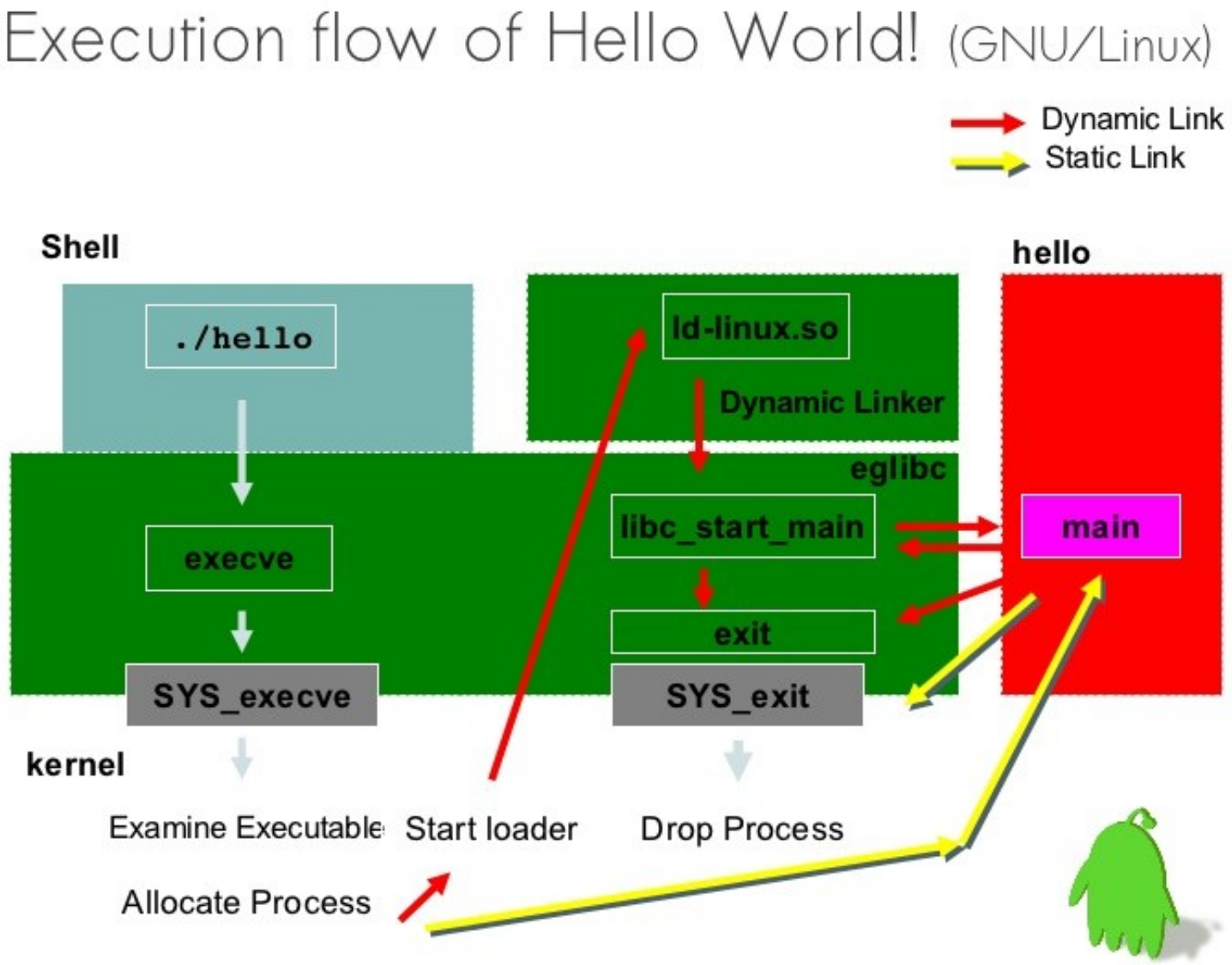



Application
System Call
Application
VFS
IPC, File System
Scheduler, Virtual Memory
Device Drivers, Dispatcher, ...
Hardware

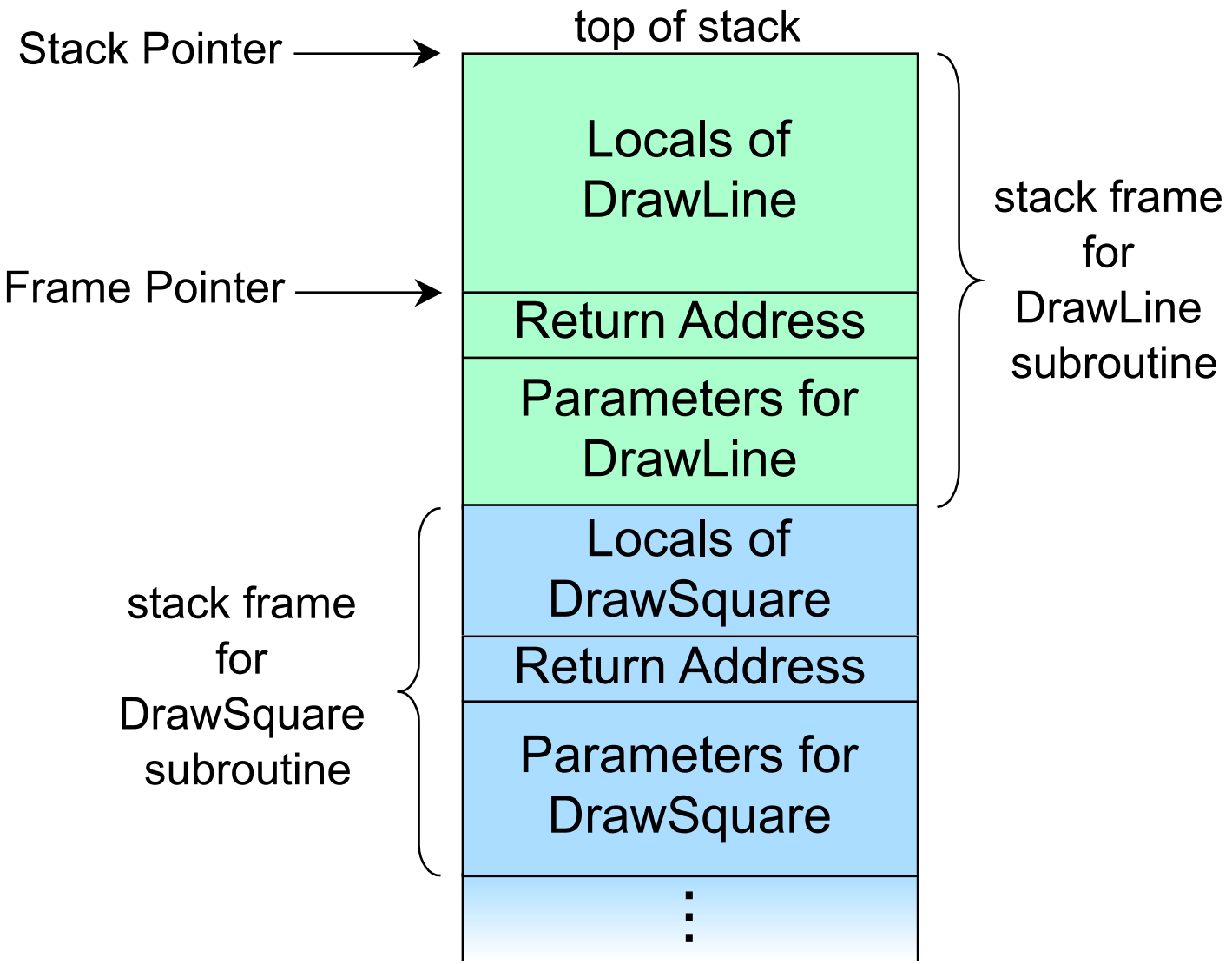
Application



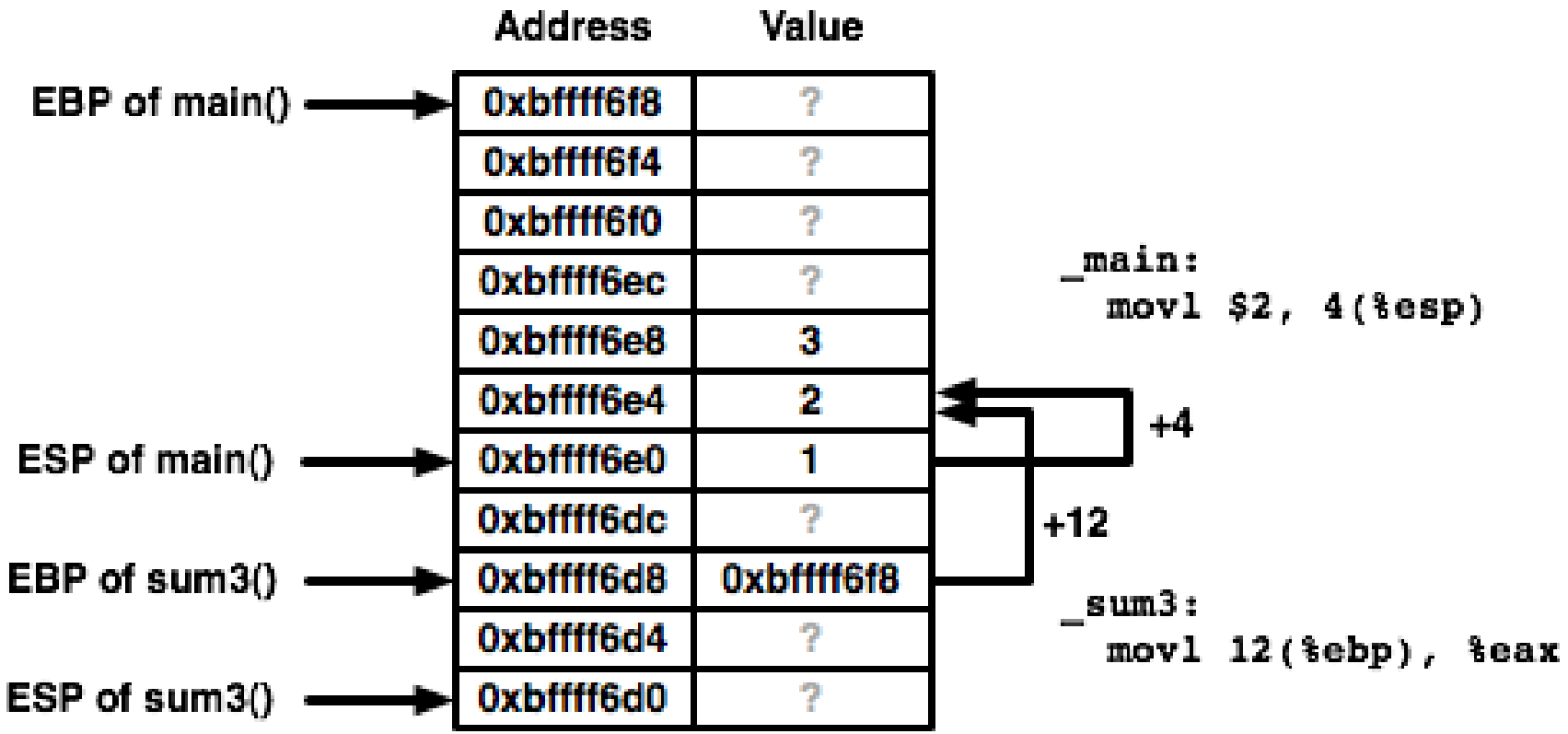
Application

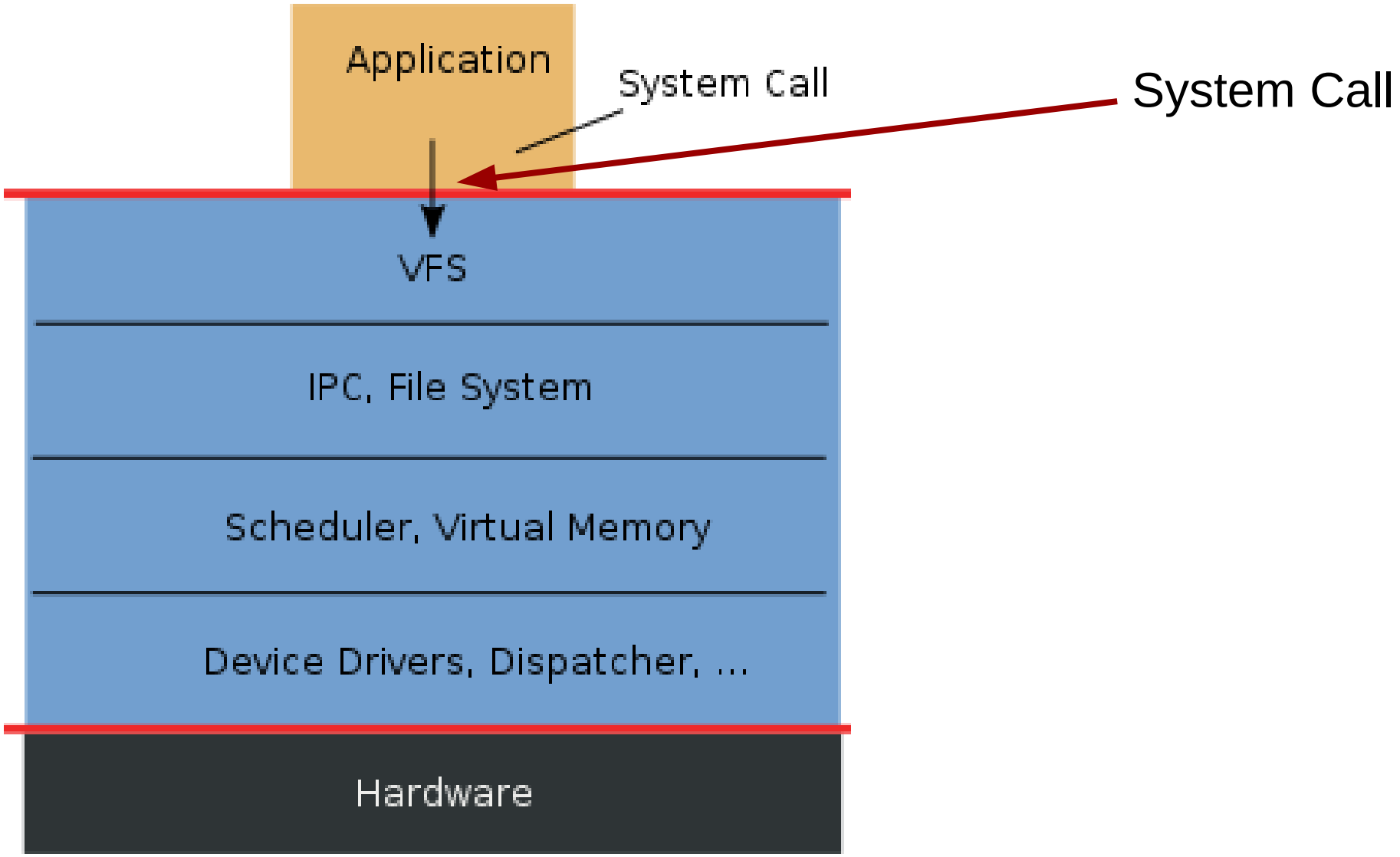


Application

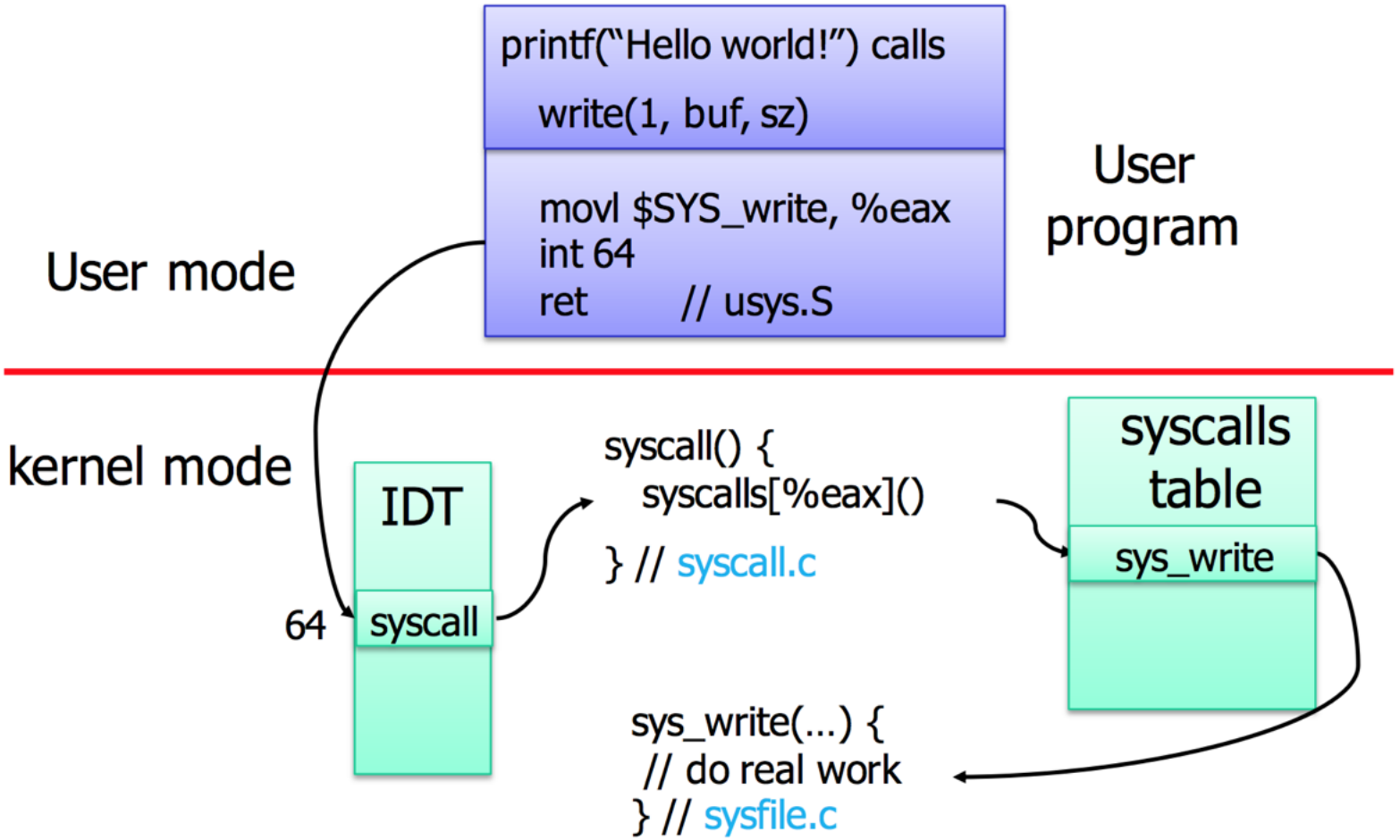


Application

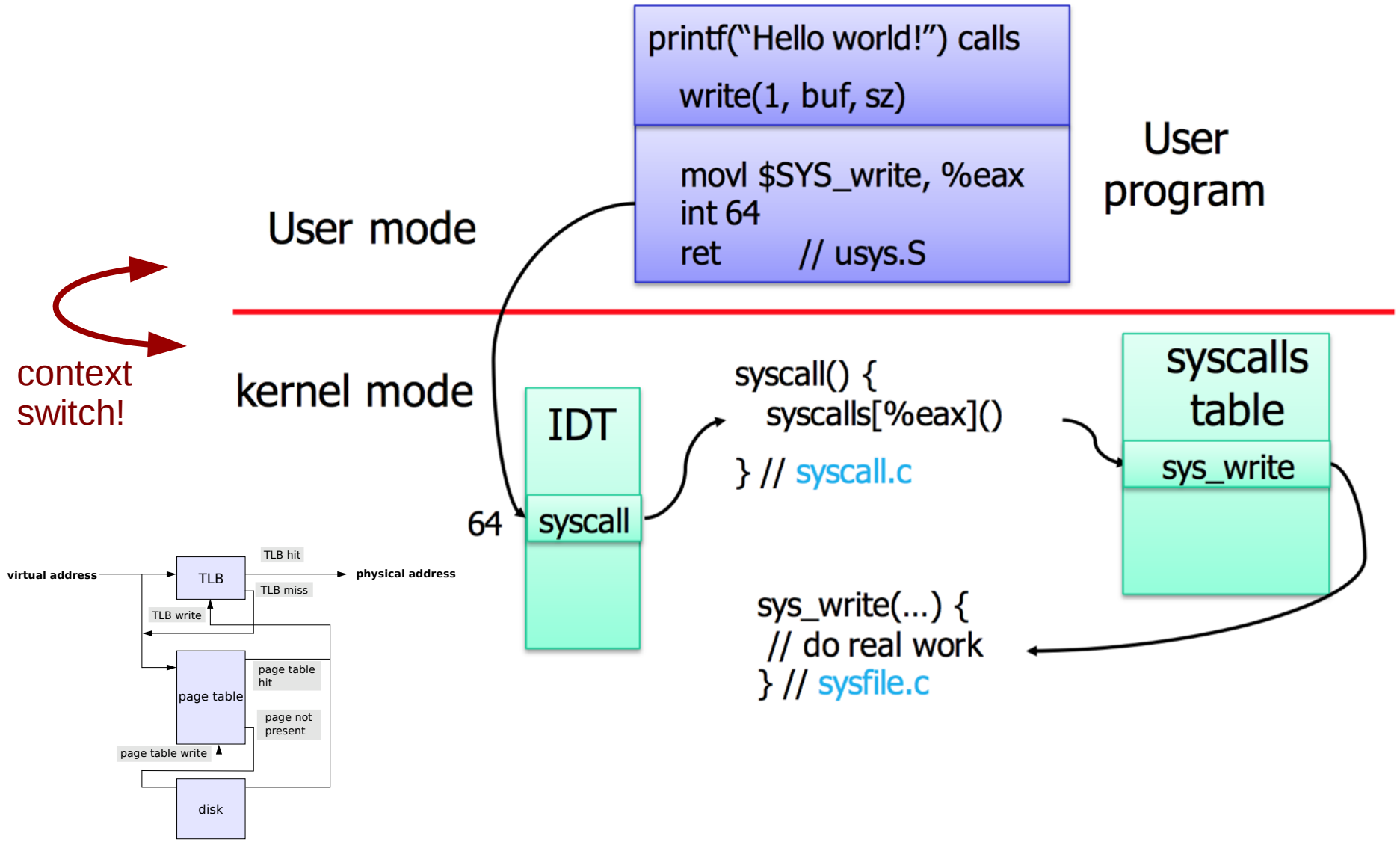


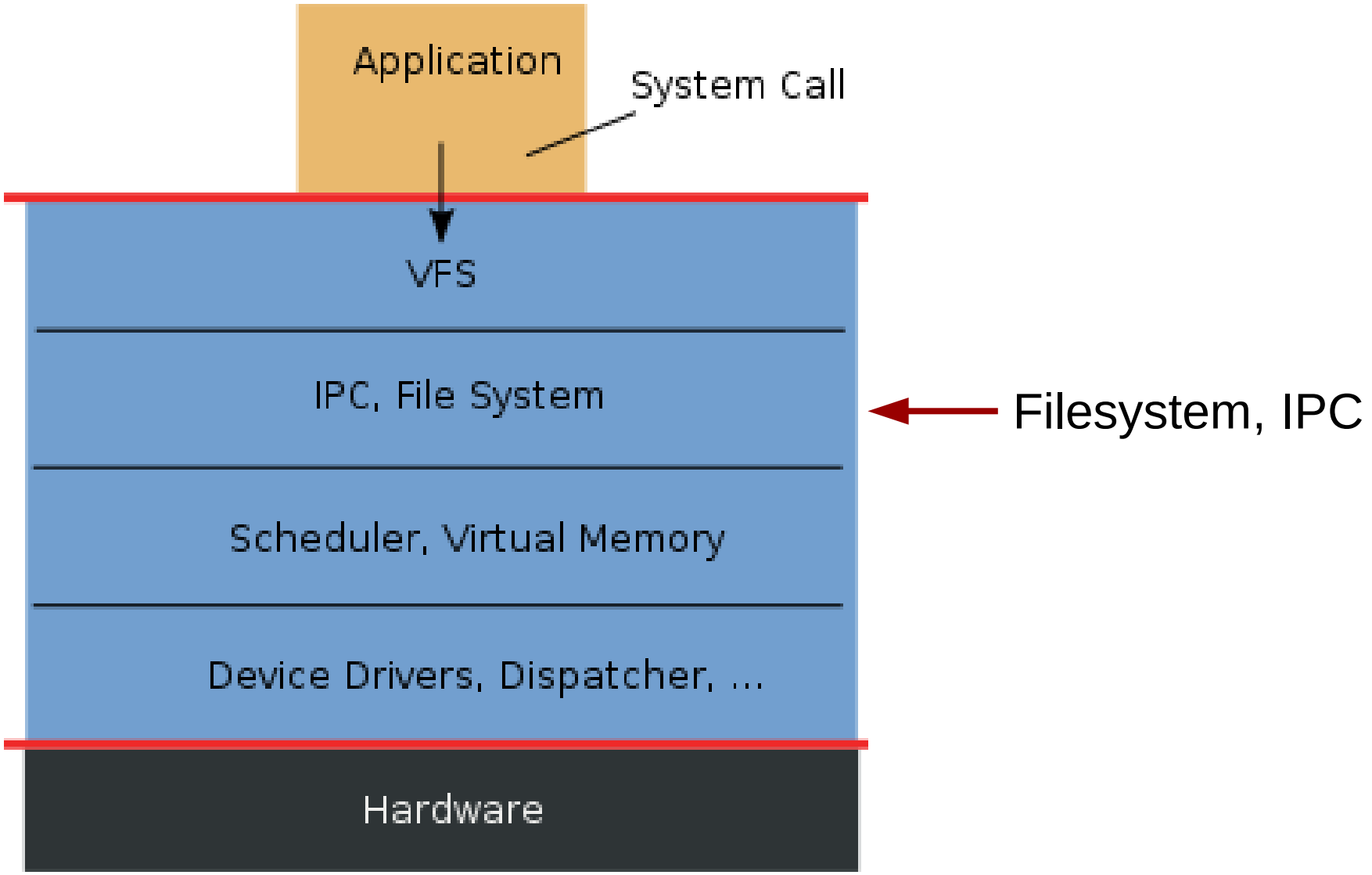

Application
System Call
System Call
VFS
IPC, File System
Scheduler, Virtual Memory
Device Drivers, Dispatcher, ...
Hardware

System call

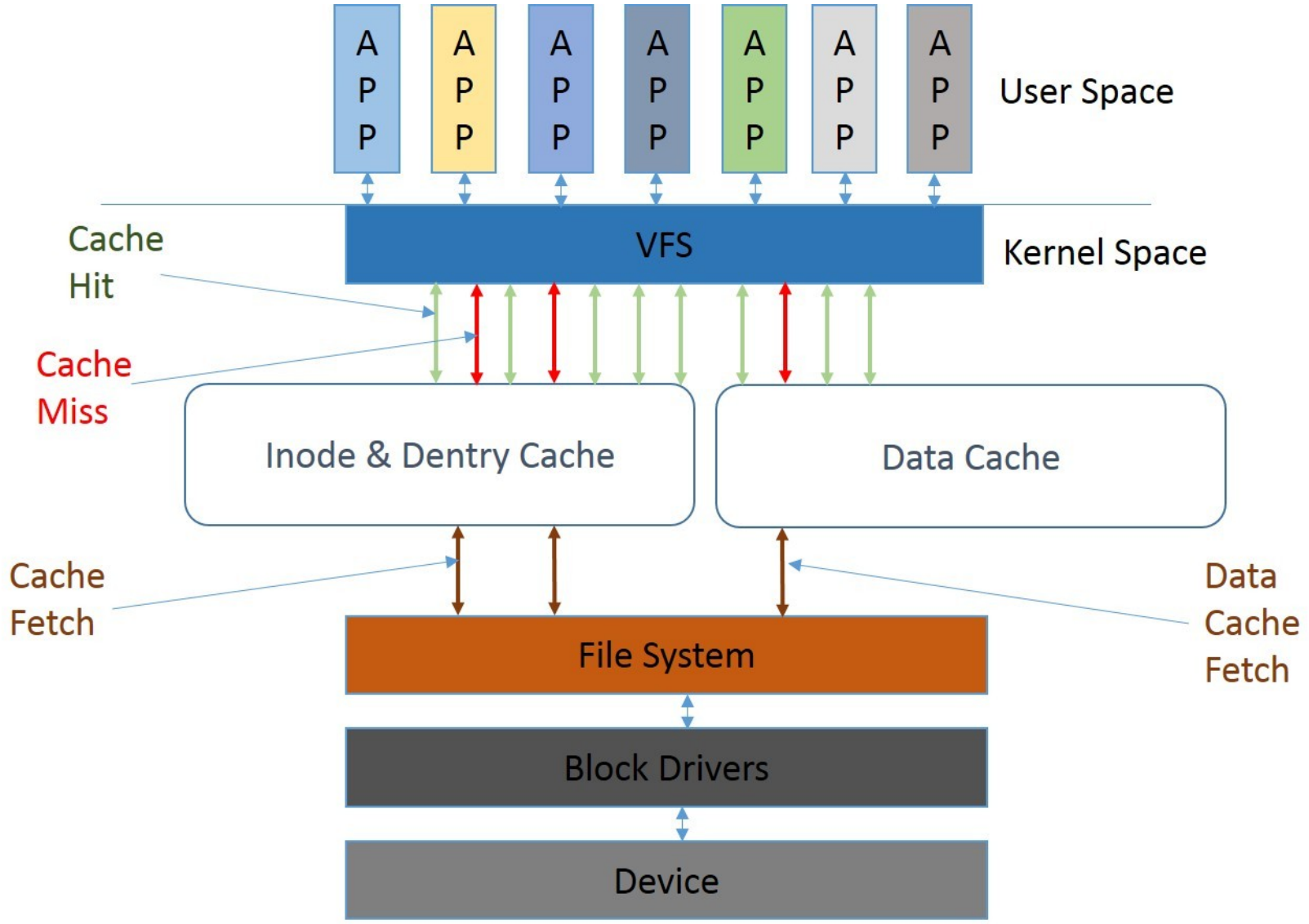

printf("Hello world!") calls
write(1, buf, sz)
movl $SYS_write, %eax
int 64
ret // usys.S
User program
User mode
kernel mode
IDT
64
syscall
syscall() {
syscalls[%eax]()
} // syscall.c
syscalls table
sys_write
sys_write(...) {
// do real work
} // sysfile.c

System call

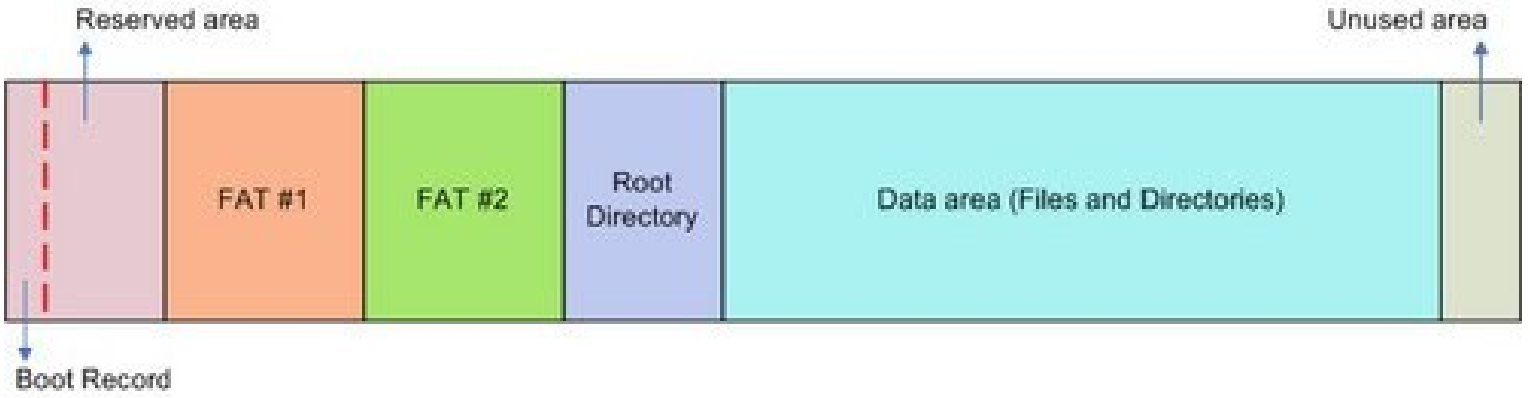

printf("Hello world!") calls
write(1, buf, sz)
movl $SYS_write, %eax
int 64
ret // usys.S
User program
User mode
kernel mode
context switch!
IDT
64
syscall
syscall() {
syscalls[%eax]()
} // syscall.c
syscalls table
sys_write
sys_write(...) {
// do real work
} // sysfile.c
virtual address
TLB
TLB hit
physical address
TLB miss
TLB write
page table
page table hit
page not present
page table write
disk


Application
System Call
VFS
IPC, File System
Scheduler, Virtual Memory
Device Drivers, Dispatcher, ...
Hardware
Filesystem, IPC

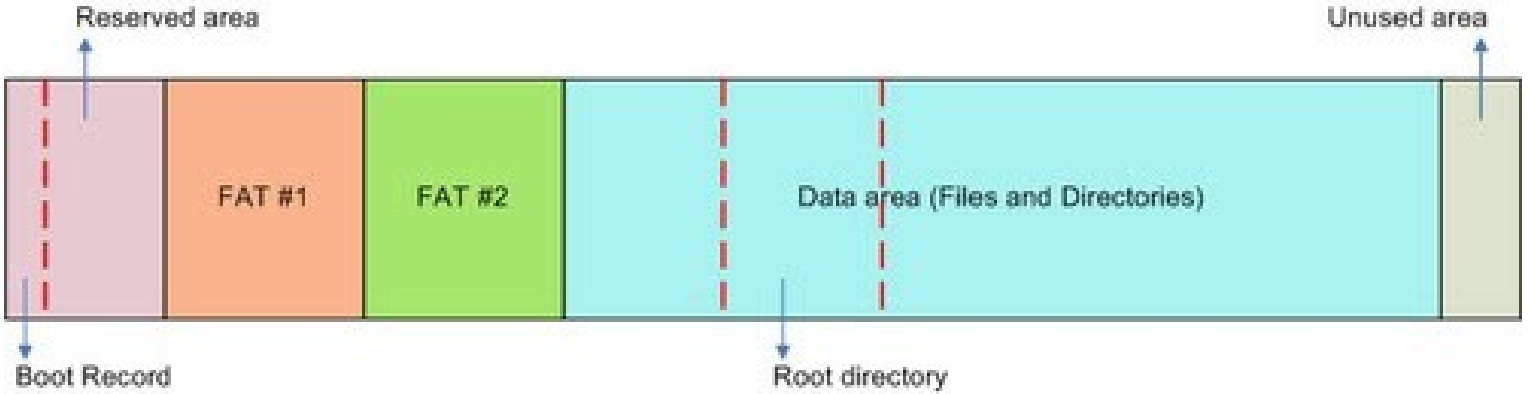
Filesystem: VFS



Filesystem (FAT32)



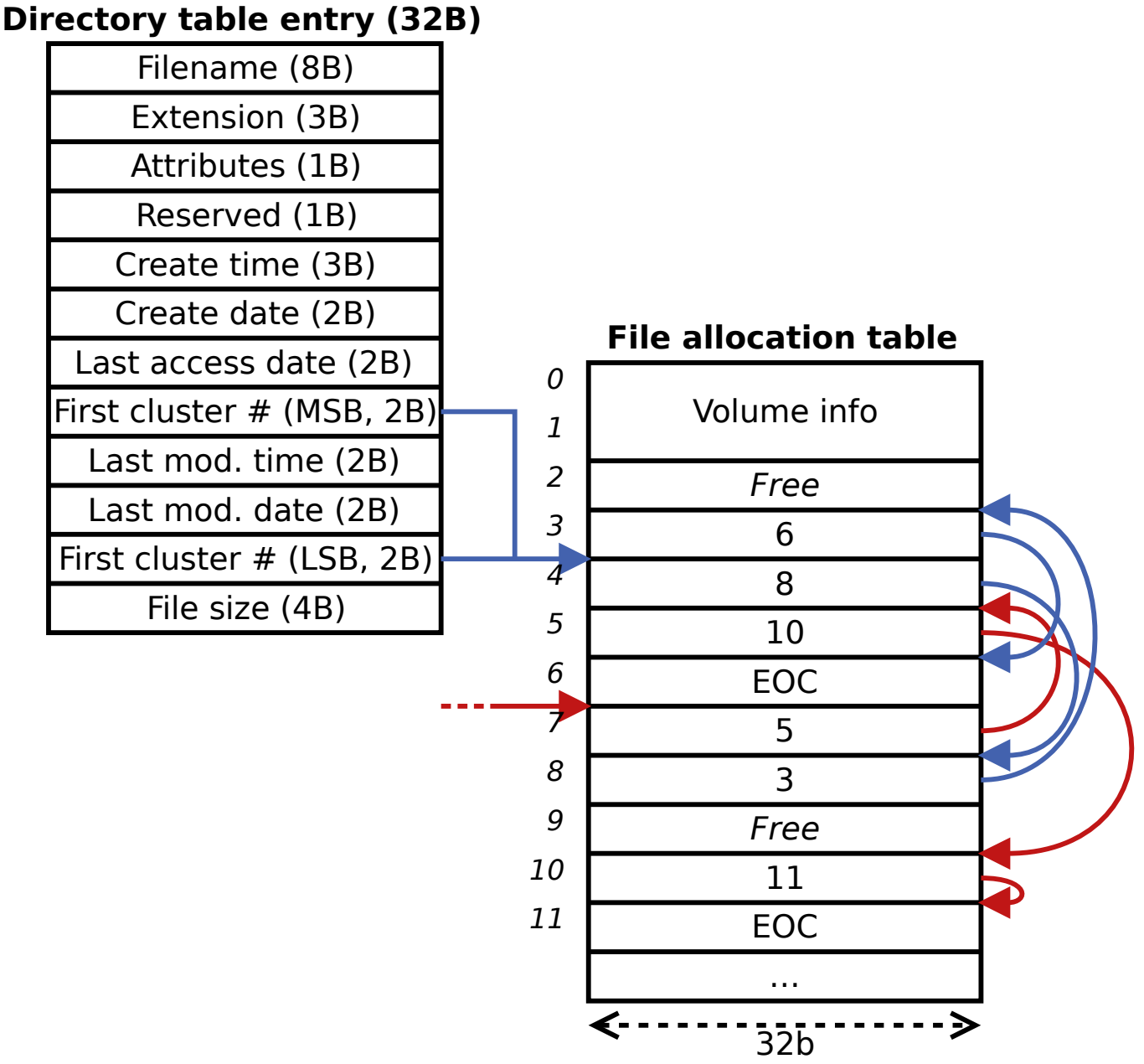
The structure of FAT16 file system



The structure of FAT32 file system

by **iprinceps**, <iprinceps@gmail.com>

Filesystem (FAT32)



Filesystem (FAT32)

XXXXXXXX	XXXXXXXX	00000009	00000004
00000005	00000007	00000000	00000008
FFFFFFFF	0000000A	0000000B	00000011
0000000D	0000000E	FFFFFFFF	00000010
00000012	FFFFFFFF	00000013	00000014
00000015	00000016	FFFFFFFF	00000000
00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000

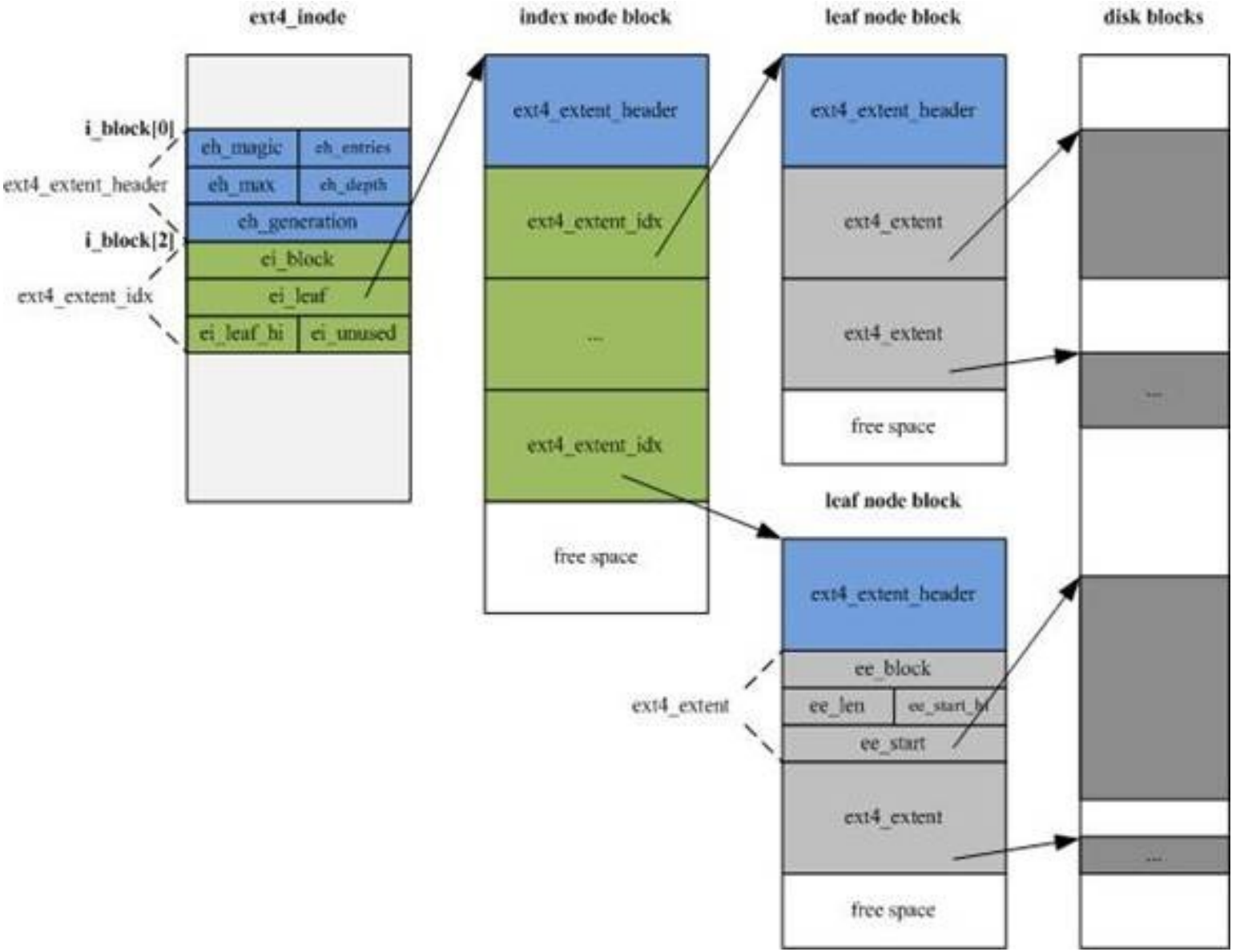
Root Directory:
2, 9, A, B, 11

File #1:
3, 4, 5, 7, 8

File #2:
C, D, E

File #3:
F, 10, 12, 13, 14, 15, 16

Filesystem (ext4)



Filesystem (ext4)

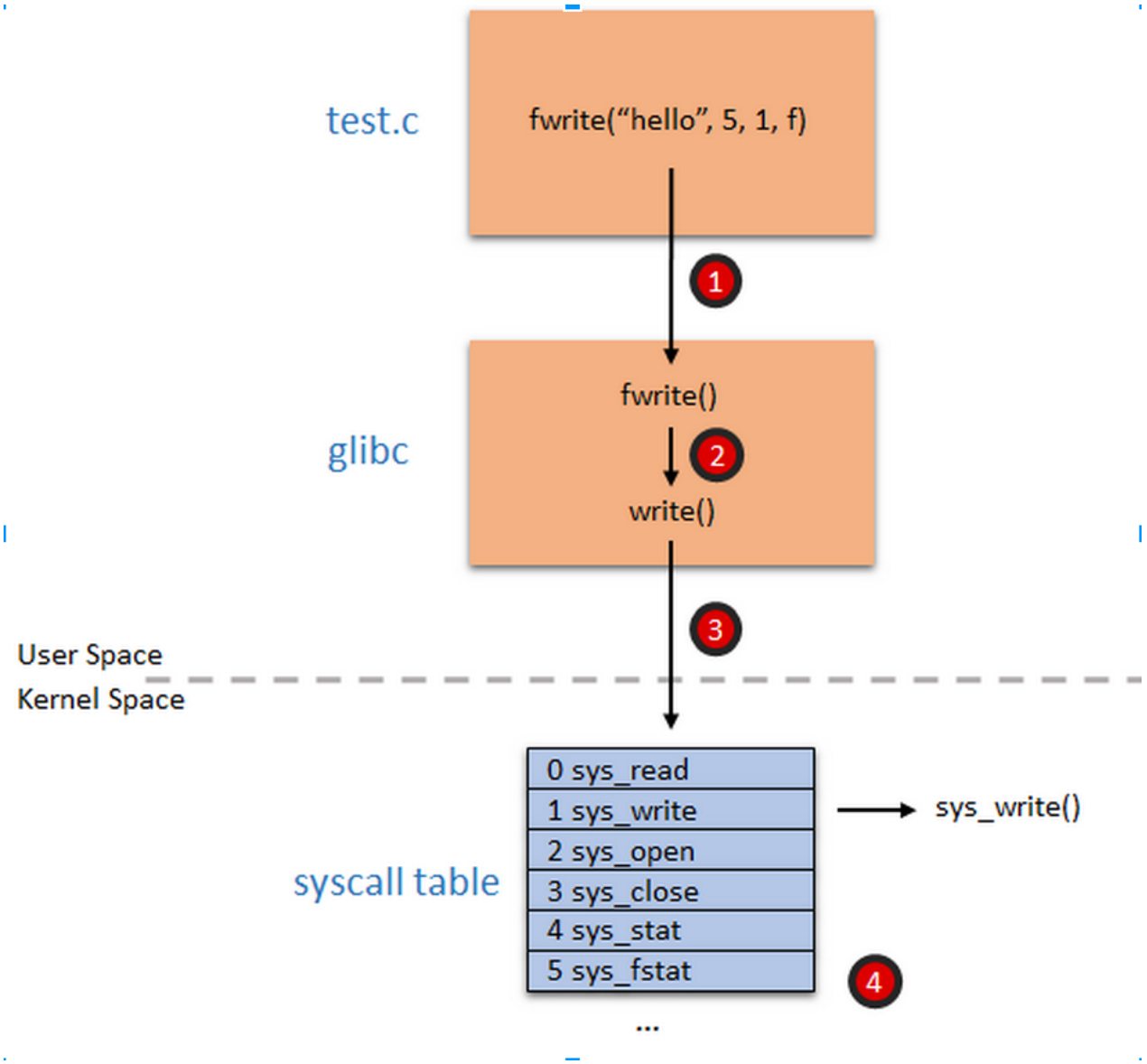
Journal				File System	
TxB	Contents		TxE	Metadata	Data
	(metadata)	(data)			
issue	**issue**	**issue**			
complete					
	complete				
		complete			
			issue		
			complete		
				issue	**issue**
					complete
				complete	

Table 42.1: **Data Journaling Timeline**

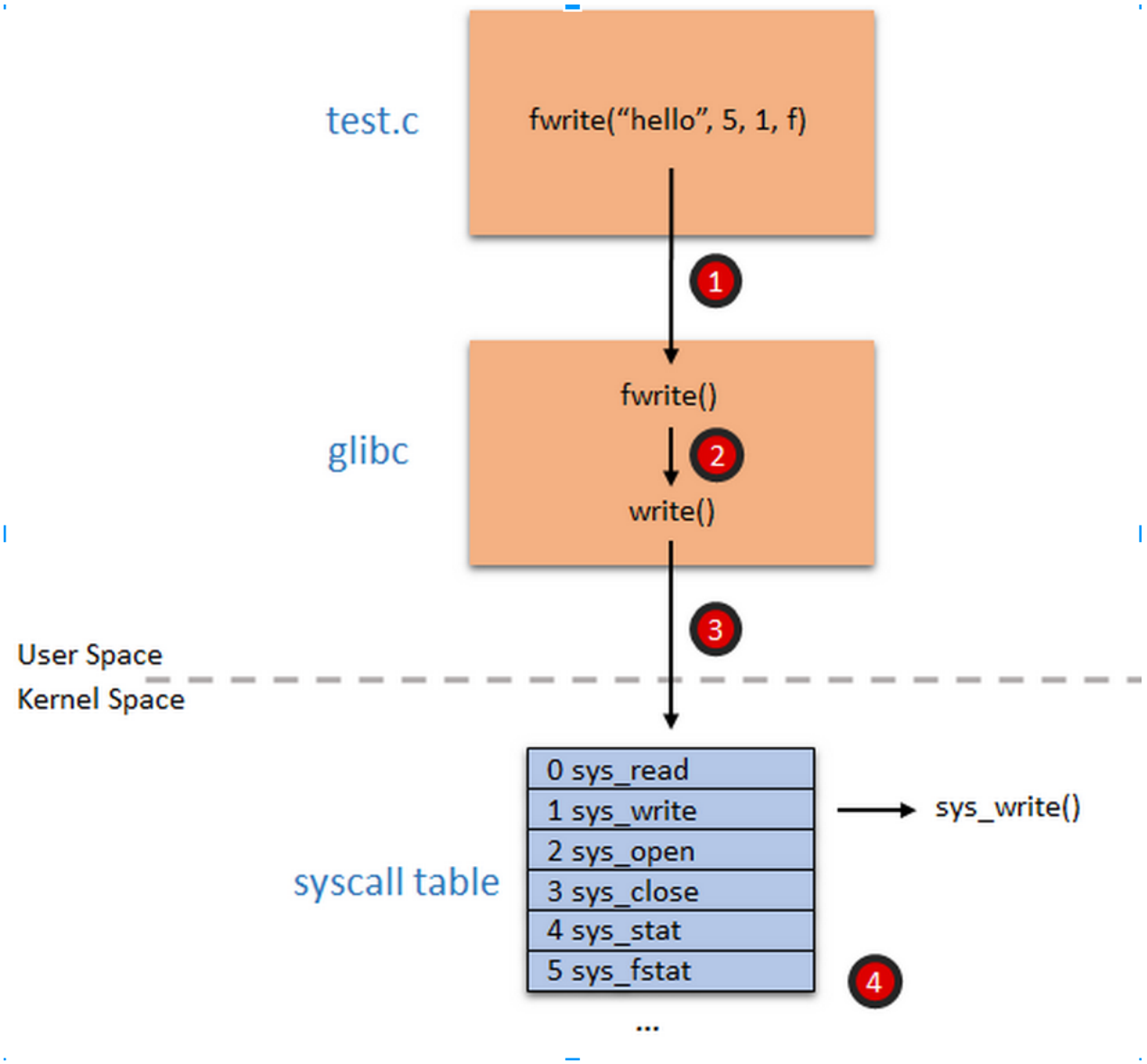
Filesystem

- Copy on Write
- Transparent compression
- SSD-cache / TRIM
- Network filesystems
- Distributed filesystems
- Pseudo file systems (wikipediaFS, PiFS, …)
- …

Filesystem

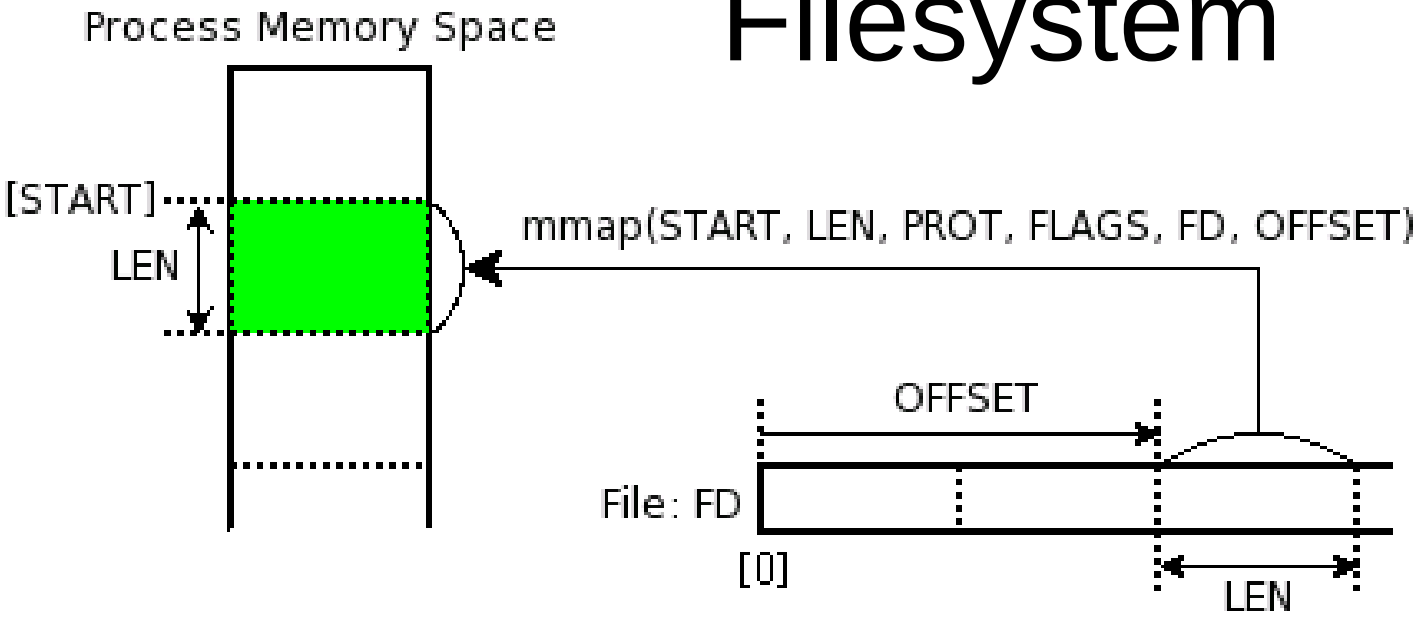

test.c
fwrite("hello", 5, 1, f)
1
glibc
fwrite()
2
write()
3
User Space
Kernel Space
syscall table
0 sys_read
1 sys_write
2 sys_open
3 sys_close
4 sys_stat
5 sys_fstat
...
sys_write()
4

Filesystem

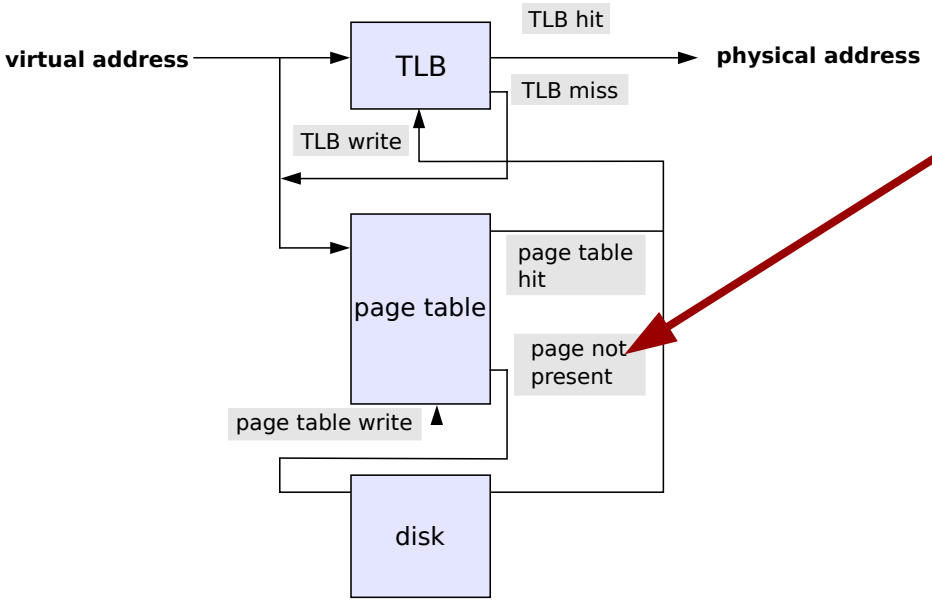


mmap() !

Filesystem



mmap() !

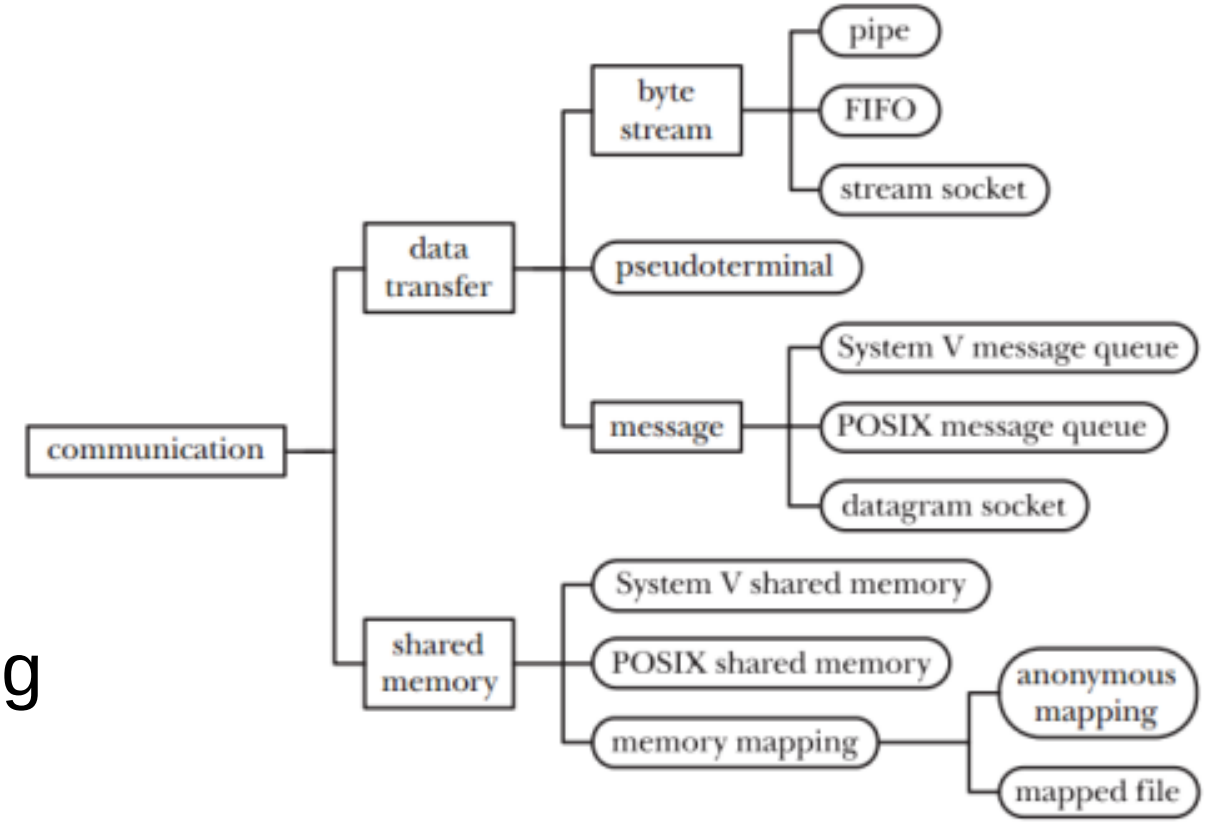


IPC

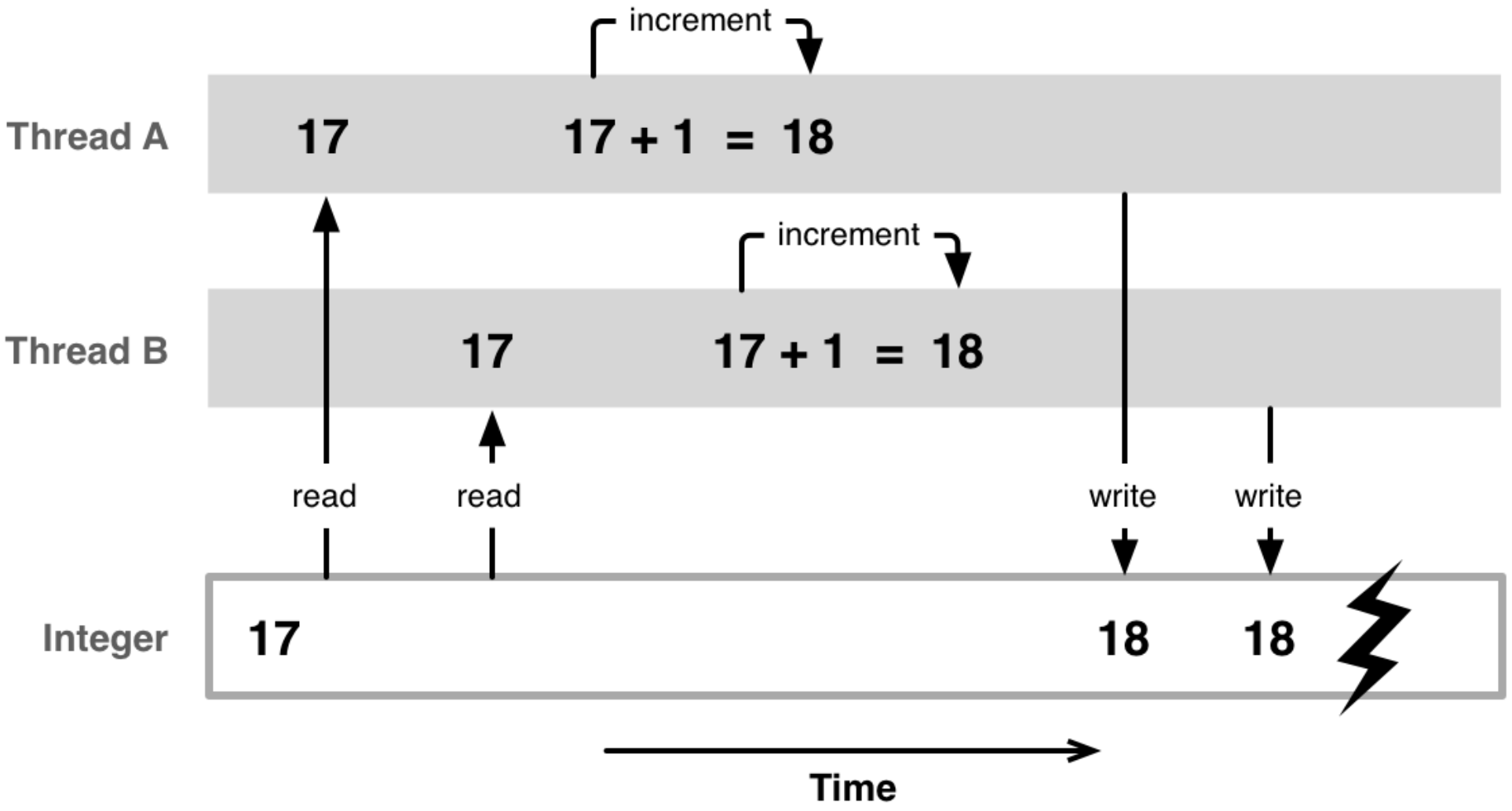
- IPC == Inter-Process Communication:
  - Files
  - Signals, Traps
  - Sockets
  - Message queue
  - Pipe
  - Shared memory
  - Message passing
  - ...

IPC

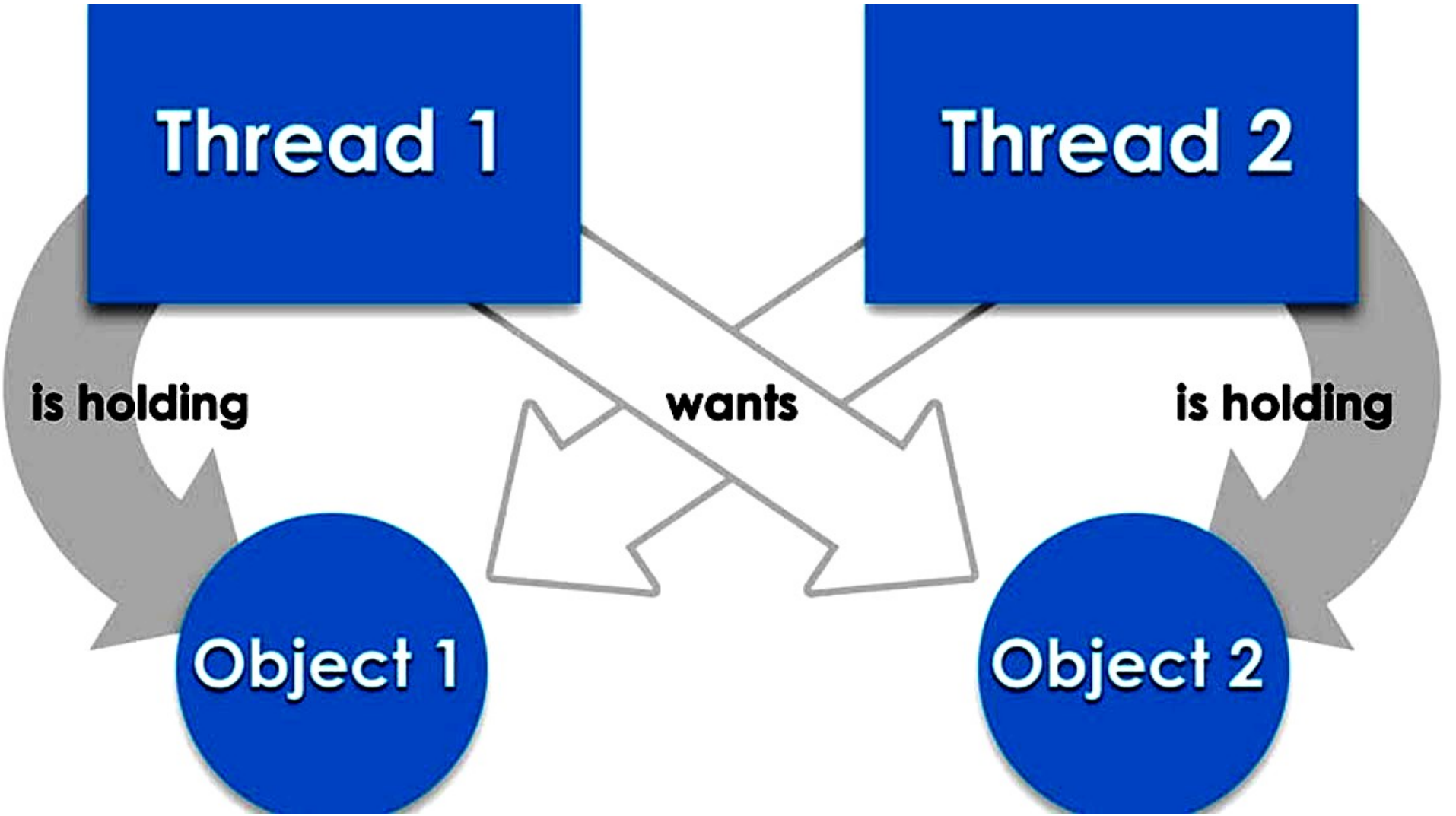
- IPC == Inter-Process Communication:
  - Files
  - Signals, Traps
  - Sockets
  - Message queue
  - Pipe
  - Shared memory
  - Message passing
  - ...



IPC: race condition



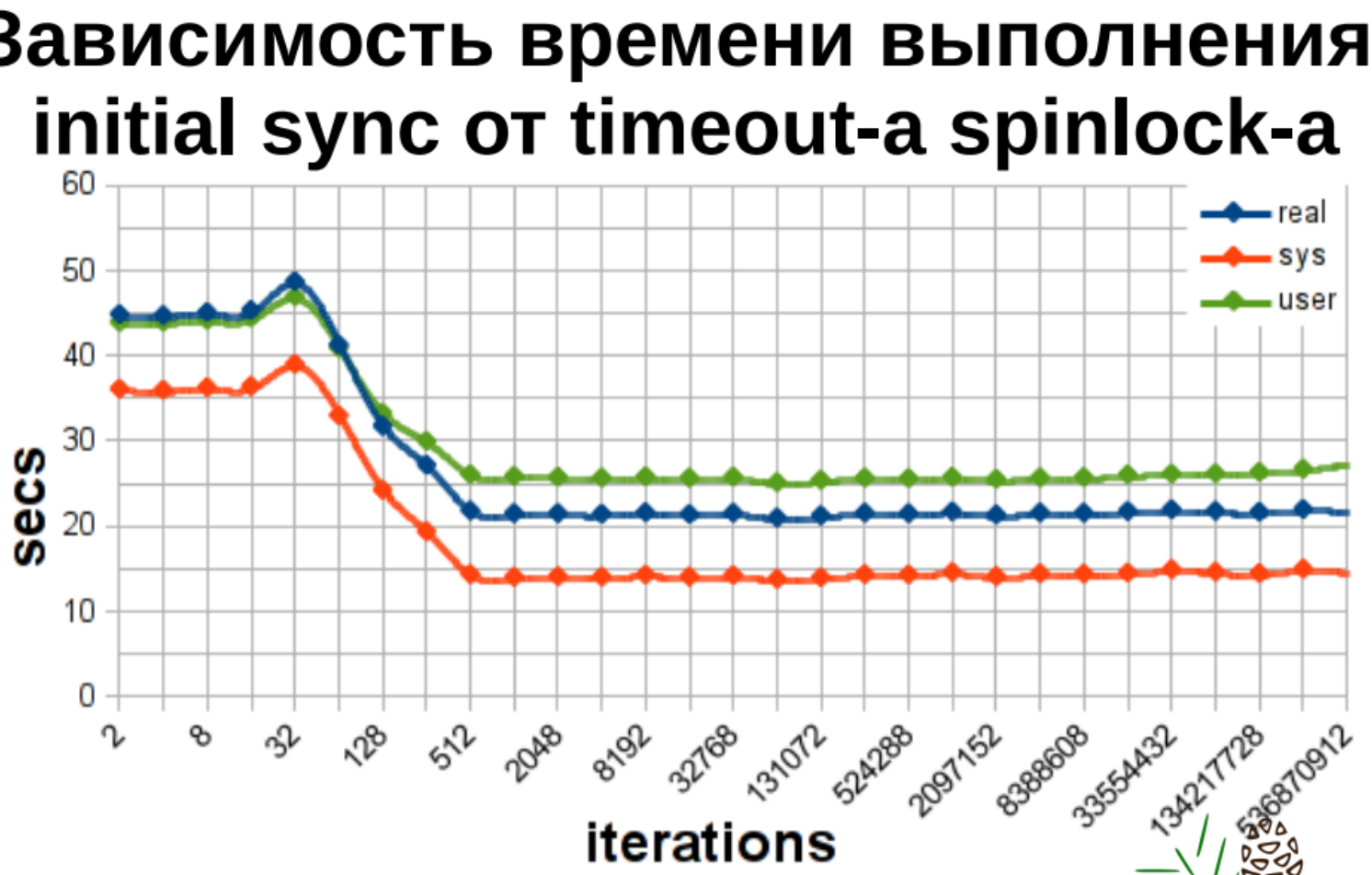
IPC: deadlock



IPC: synchronization

Technique	Description	Scope
Per-CPU variables	Duplicate a data structure among CPUs	All CPUs
Atomic operation	Atomic read-modify-write instruction	All
Memory barrier	Avoid instruction re-ordering	Local CPU
Spin lock	Lock with busy wait	All
Semaphore	Lock with blocking wait (sleep)	All
Seqlocks	Lock based on access counter	All
Local interrupt disabling	Forbid interrupt on a single CPU	Local
Local softirq disabling	Forbid deferrable function on a single CPU	Local
Read-copy-update (RCU)	Lock-free access to shared data through pointers	All

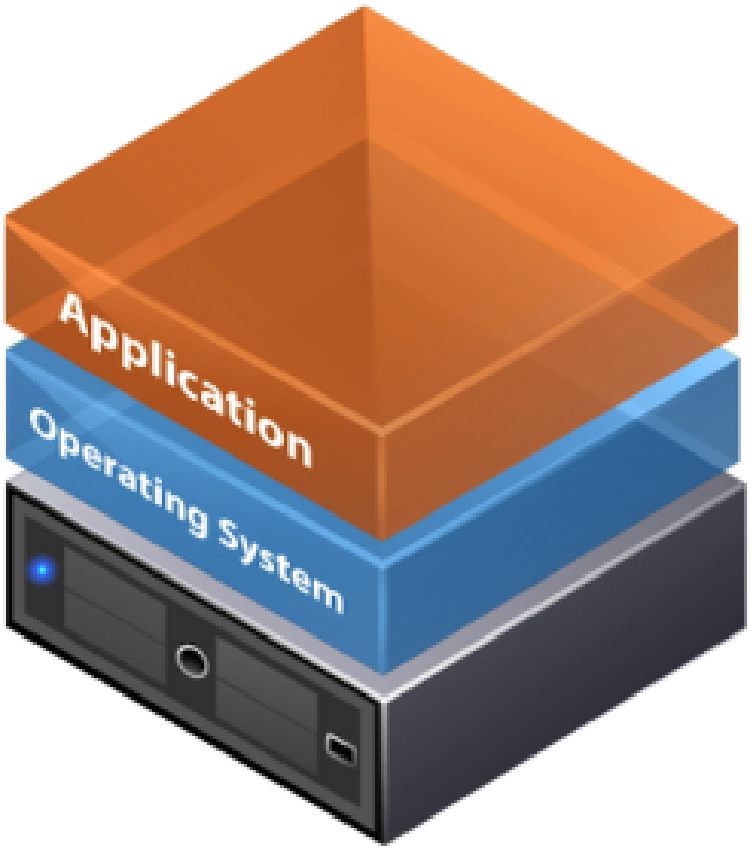
IPC: synchronization

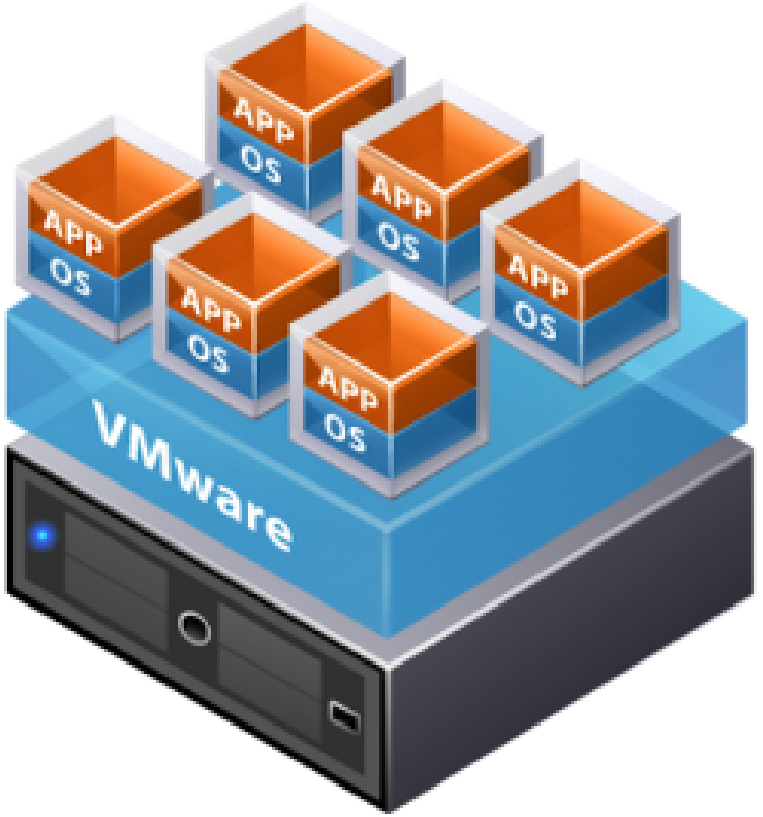


LVEE 2014
Linux Vacation / Eastern Europe

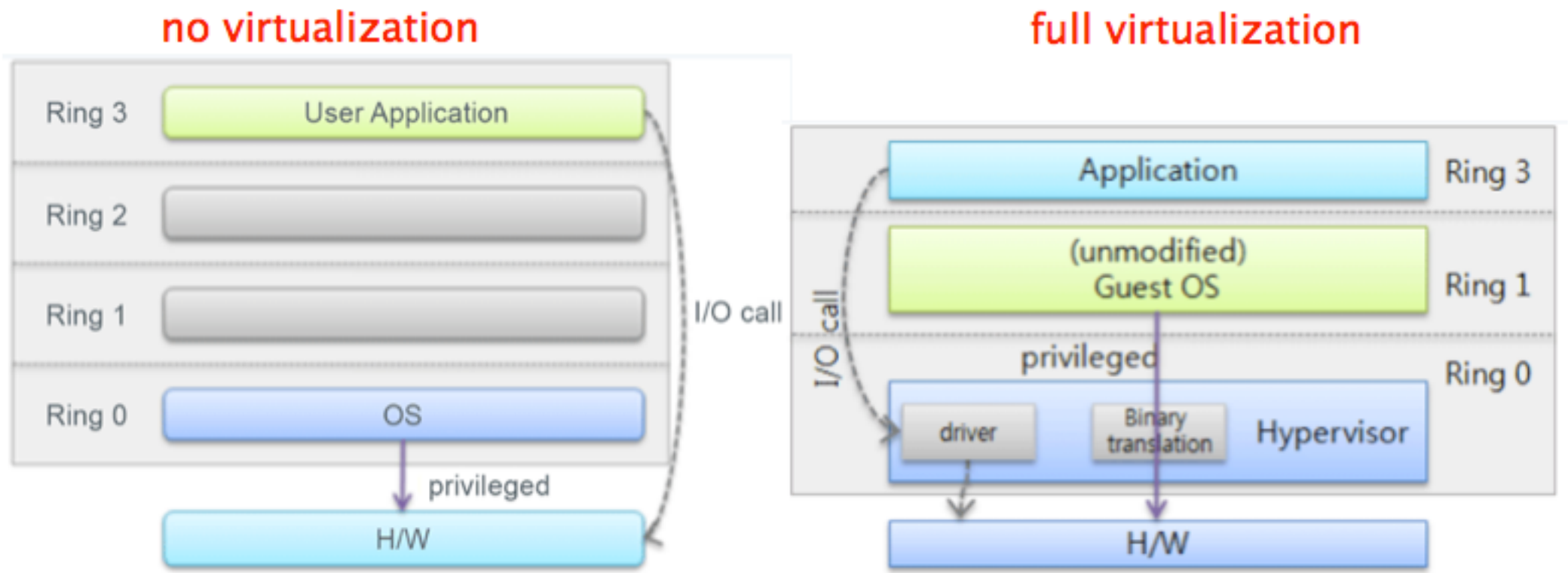
For dessert: virtualization, containers

vmx:

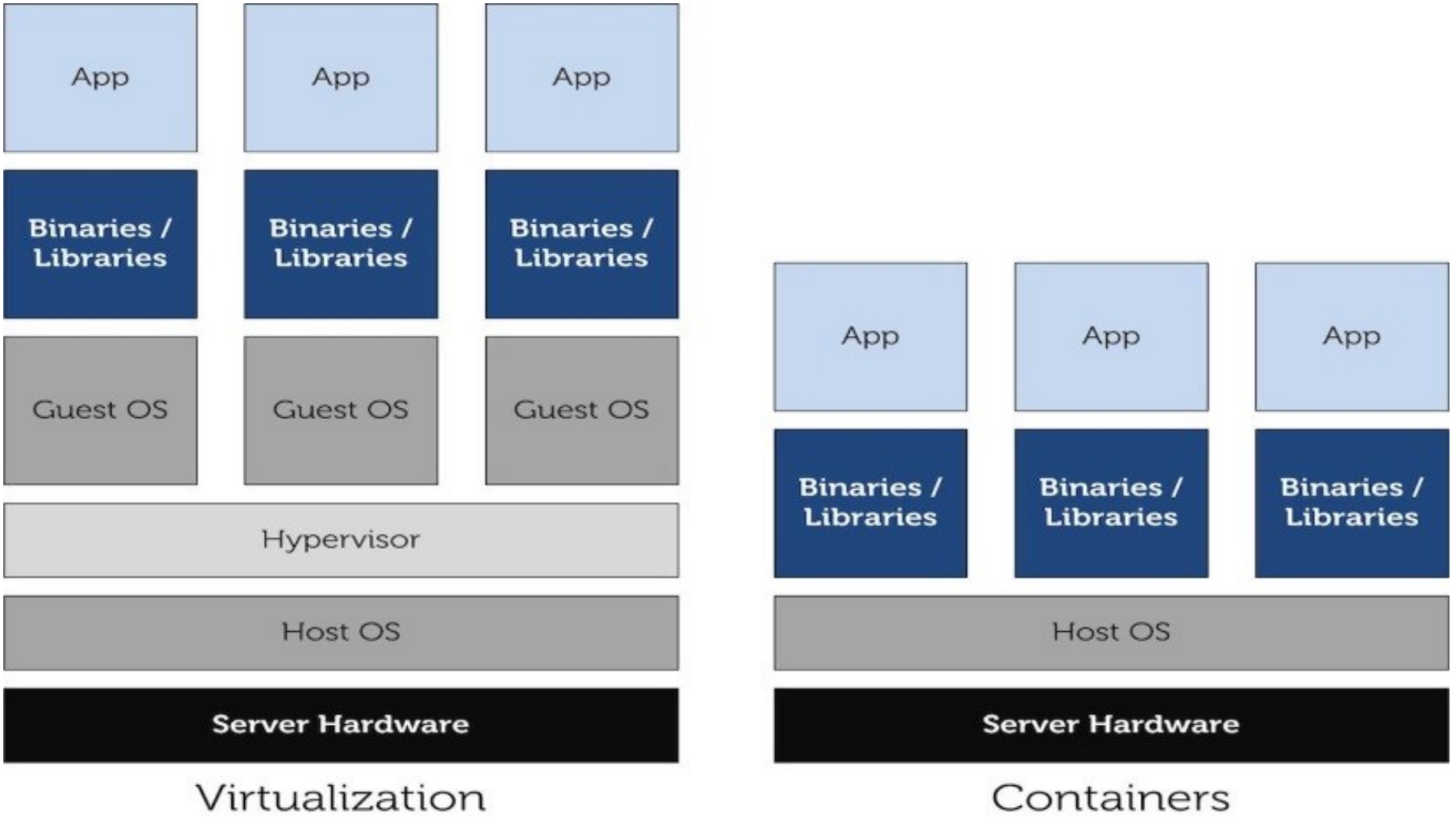




Virtualization



Containers


App
App
App
Binaries / Libraries
Binaries / Libraries
Binaries / Libraries
Guest OS
Guest OS
Guest OS
Hypervisor
Host OS
Server Hardware
Virtualization
App
App
App
Binaries / Libraries
Binaries / Libraries
Binaries / Libraries
Host OS
Server Hardware
Containers

Q&A