

BoVaZa

# Exception Handling

Dual stack architecture

Conditional stack switching

Conditional stack allocation

User mode enabling

Re-entrant exceptions

# Timer

Uses COUNT and COMPARE registers

Enlarge to 64 bits

Constant speed

Timer start -> calculate its COMPARE value

Next timer to COMPARE register

# Physical memory allocator

Two bitmaps

Two structures

Physical memory data structure directly in  
frame blocks

Optional parallel frame zeroing

# Memory allocation

Growing kernel heap – purging physical memory allocator

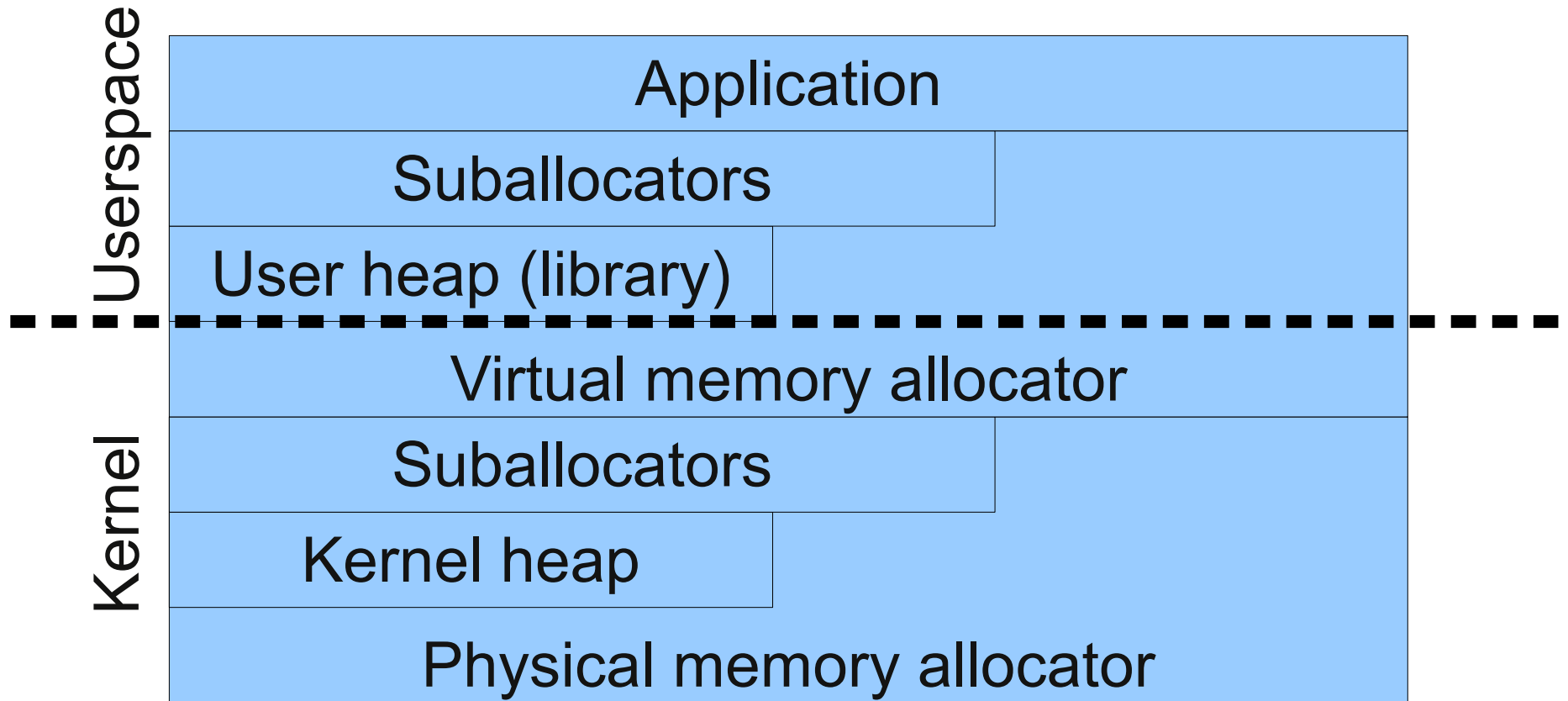
General allocator vs. suballocators

Coalescing heap – variation to buddy system

Small blocks suballocator

Thread local suballocator (under development)

# Memory allocation structure



# Virtual memory management

TLB driver

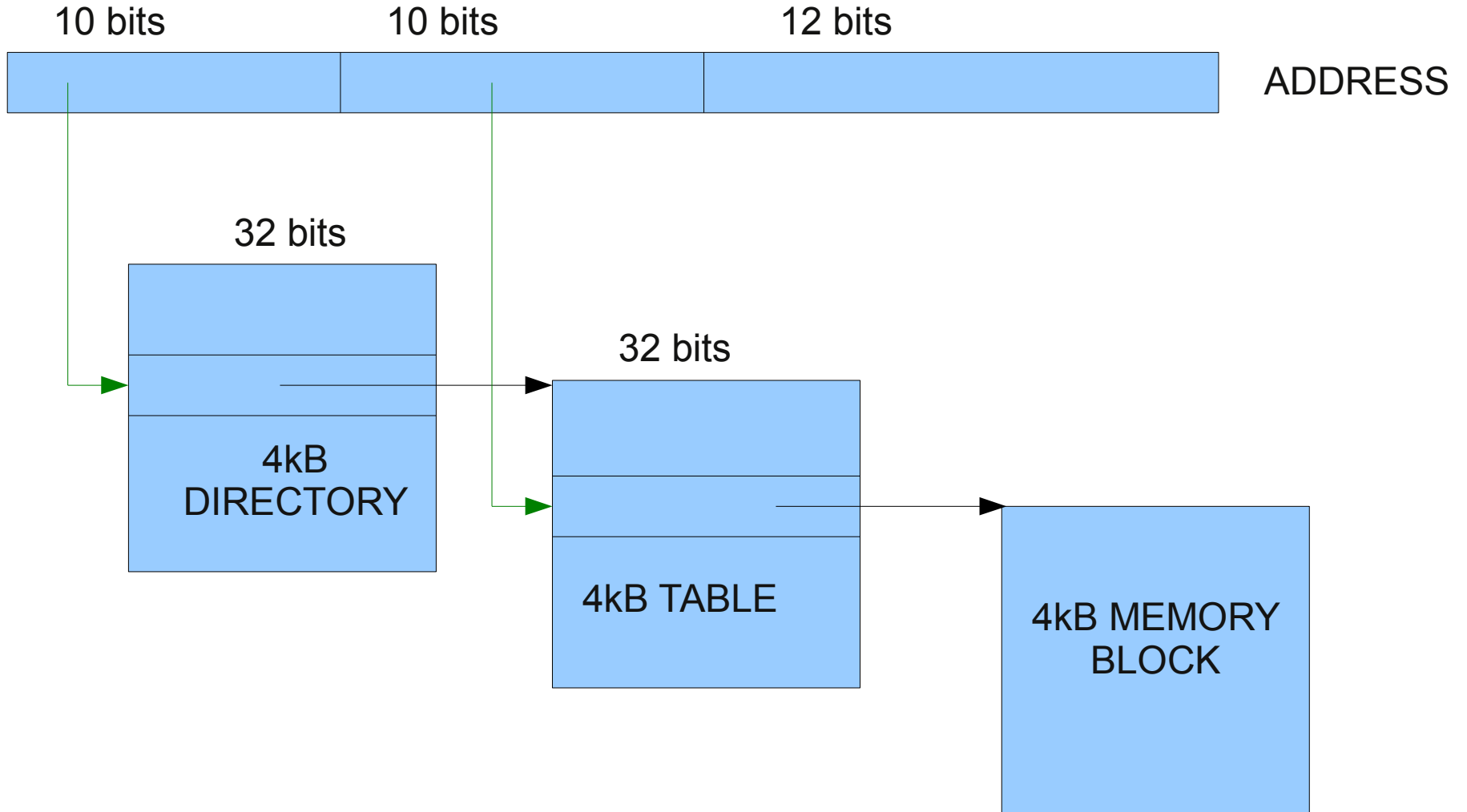
Mapping features

- Two level virtual address translation table

- Second kernel internal VMT table

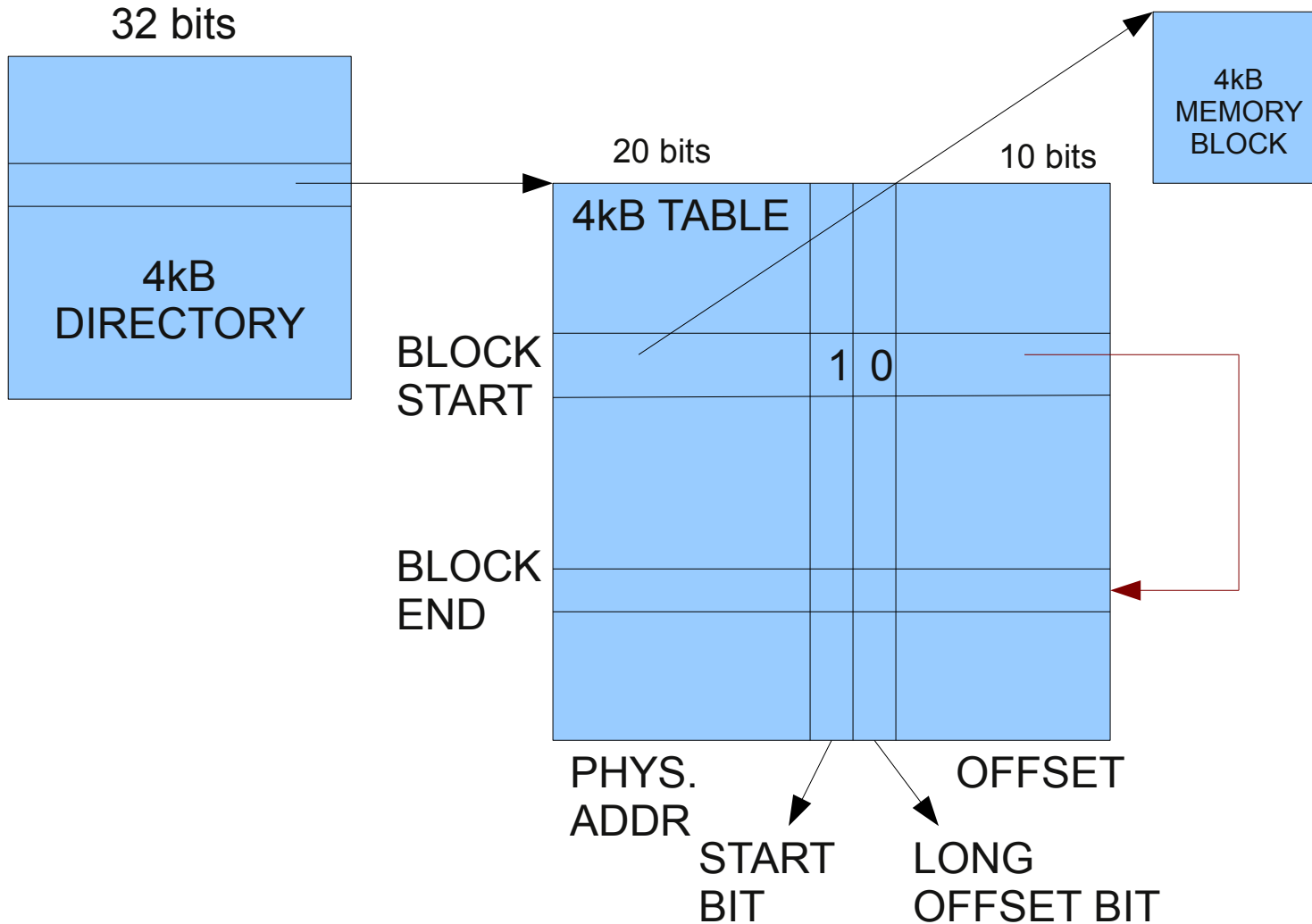
Possible to map previously allocated memory

# Virtual memory tables

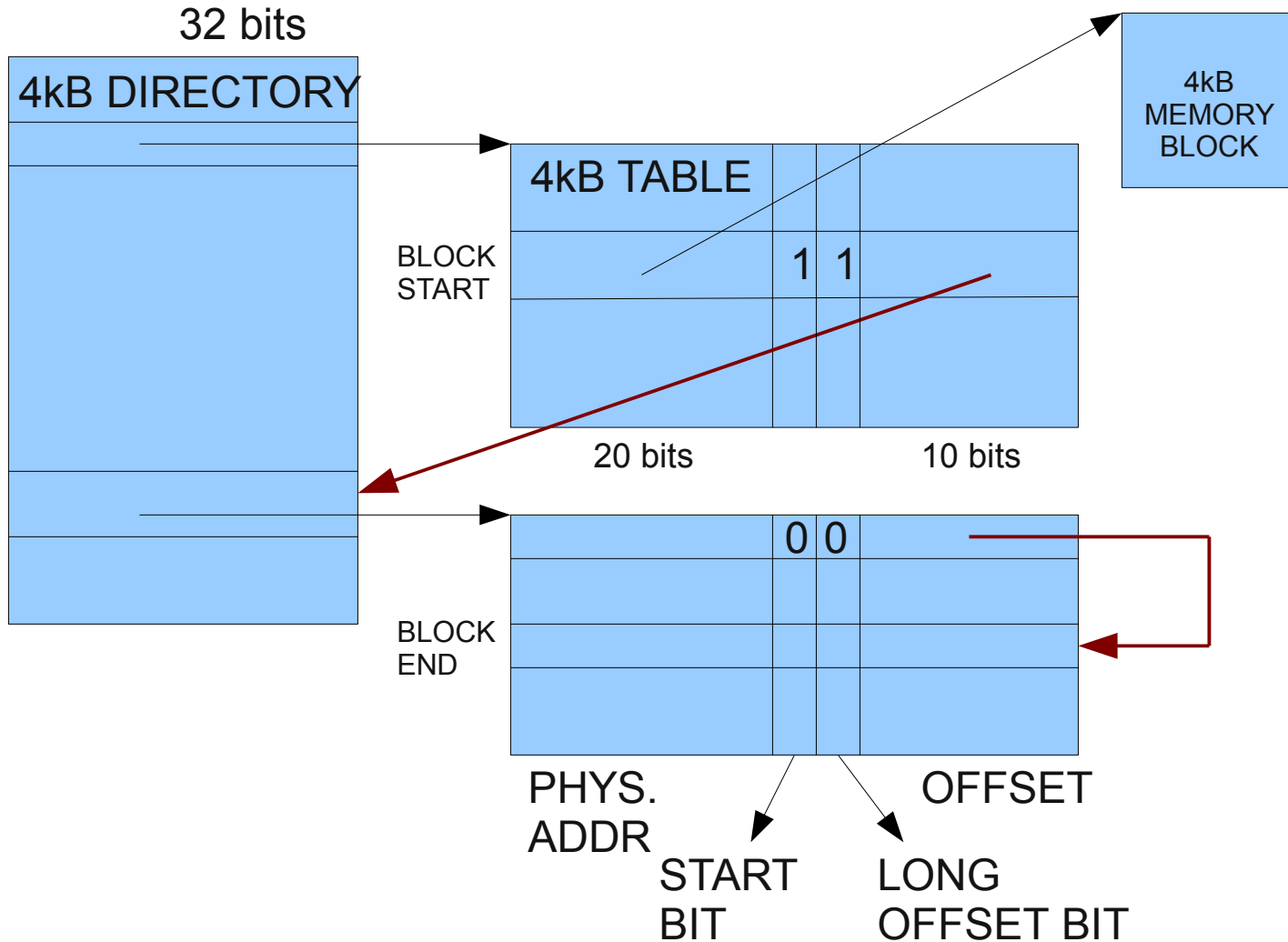




# VM – short blocks



# VM – long blocks



# Stopwatch

Measurement units: milliseconds or counts?

Stopwatch's methods are not counted

Automatic pause on context switch

# Userspace

Executable file format (also in context of physical memory management)

Callgate, format of syscalls, register based argument passing

Handles, user can not destroy kernel structures by passing wrong argument

Unkillable active threads in syscall

# Userspace (continued)

I/O operations: device mastering

Implementation through non-blocking buffers

Synchronous/asynchronous memory  
unmapping

Possible use for message passing and  
interprocess communication