

# Intel x86 Memory Architecture

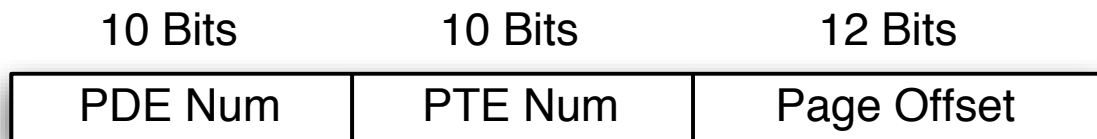
- 2-Level Page Table

- 4KB Page Size

- 32 bit addresses

- PDE/PTE of 32 bits

## Virtual Address Format



## PDE/PTE Format



# Translation

Describe the result of accessing the following virtual addresses:

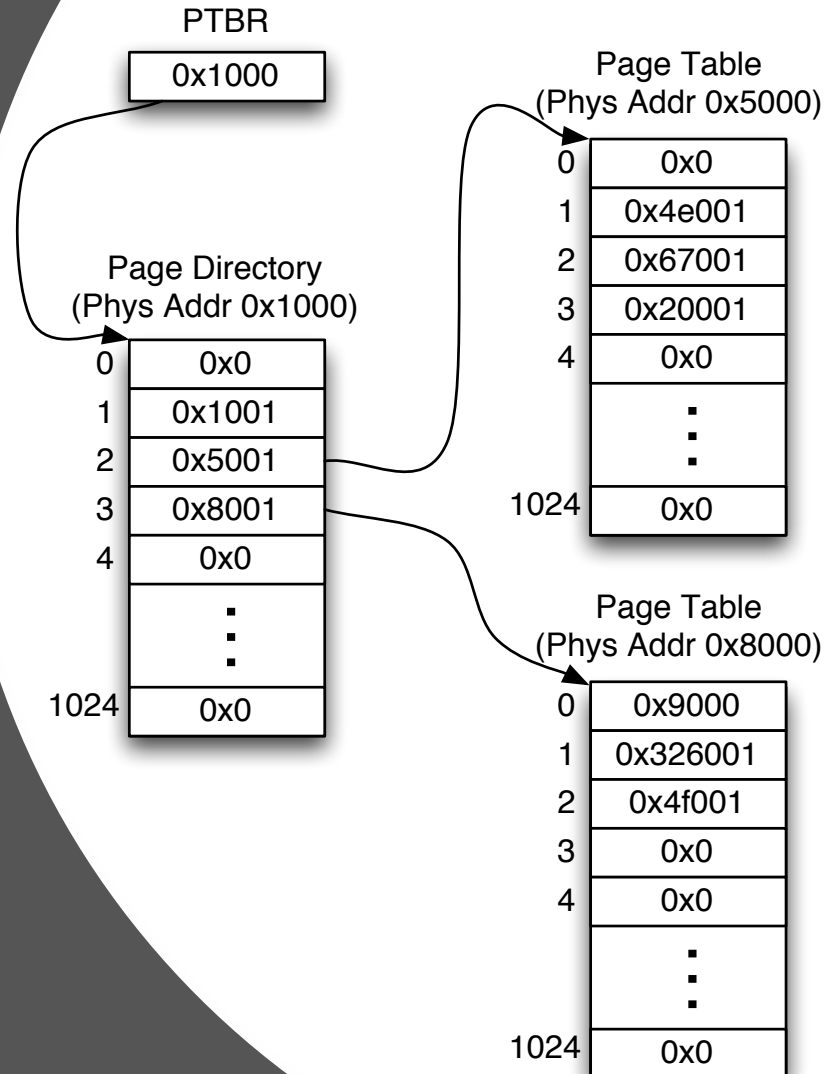
0x0

0x00803024

0x00c00136

( $2^{22} == 0x400000$ ,  
 $2^{12} == 0x1000$ )

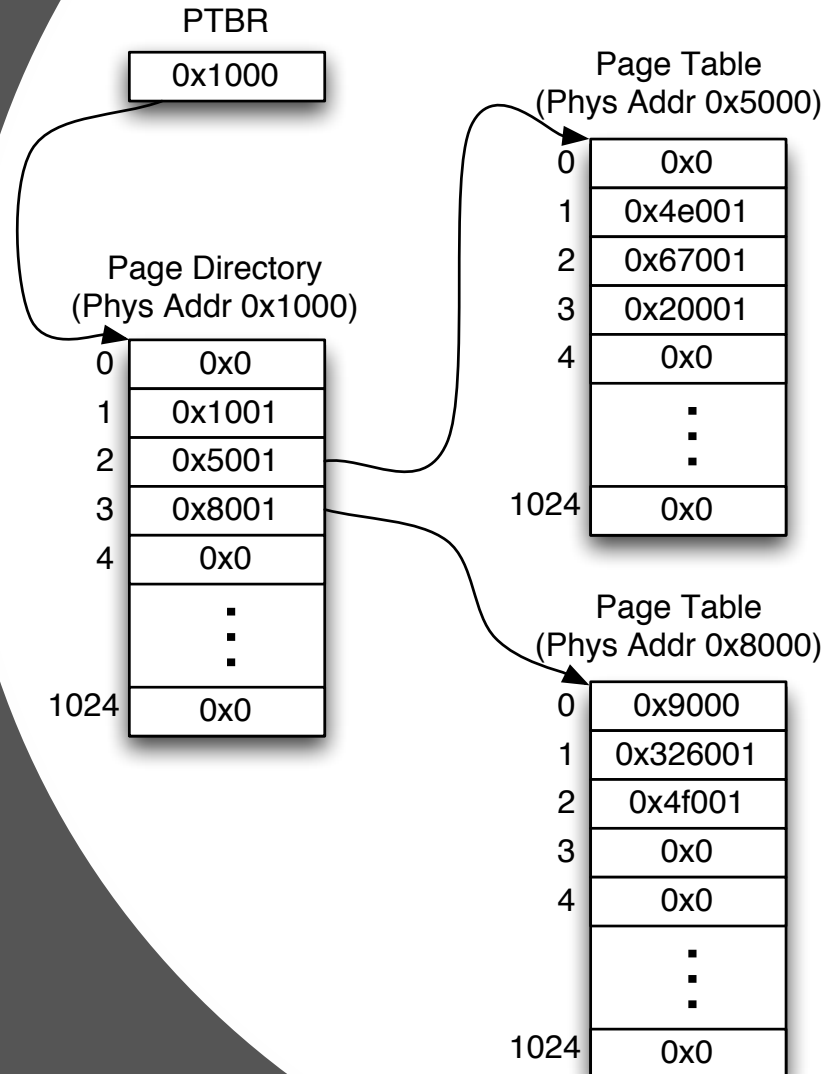
Answers: fault, 0x00020024, fault



# Translation

What is the data stored at virtual address 0x00402004?

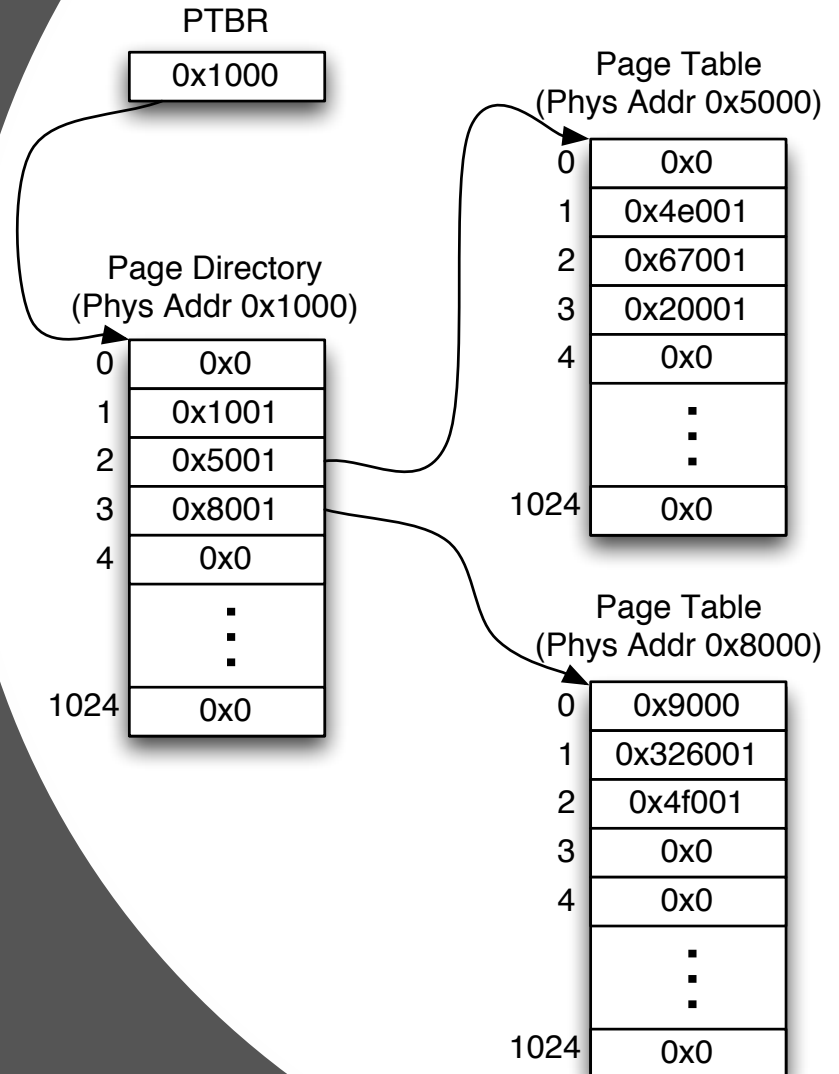
Answer: 0x0004e001



# Translation

List the physical frames that this address space has direct access to. Is this address space properly isolated from accessing any other frames?

Answers: 0x1000, 0x5000, 0x8000, 0x326000, 0x4f000, 0x200000, x67000, 0x4e000.  
Ignoring kernel/user bits and write protection, the page tables have been made accessible to the address space (virtual addresses 0x00400000-0x004ffffff), so a process running in this address space could map-in any physical frame it wanted to.



# TLB Hit Rates

Consider a x86 program consisting of 33% load/store instructions. How many extra memory accesses per instruction executed does this program need when the TLB has a 0%, 95%, or 100% hit rate?

Answers: 1.33 base memory accesses per instruction, 100%=0 extra, 0%=2.66 extra, 95%=0.05\*2.66 extra

# TLB Hit Rates

$$EAT = P_{hit} T_{miss} + (1 - P_{hit}) T_{miss}$$

L1 cache hit: 1ns

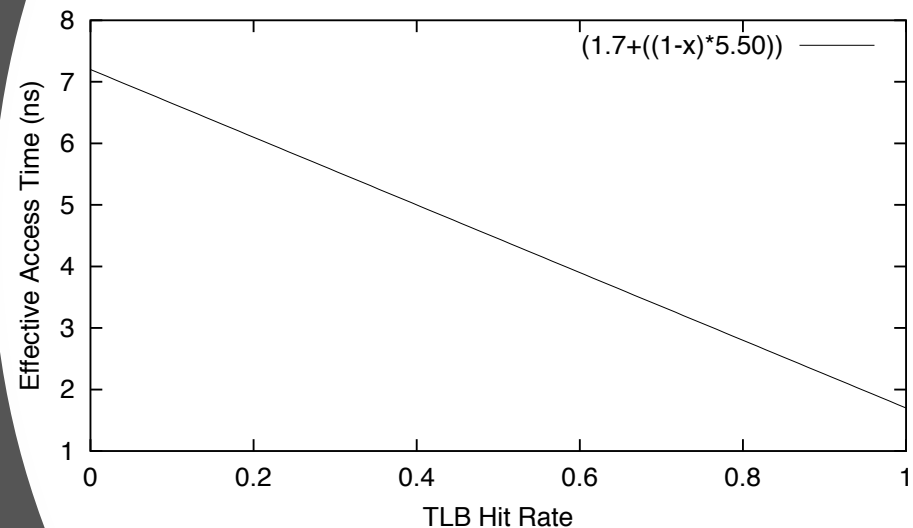
L1 cache miss: 8ns

TLB cache hit: 0ns

L1 hit rate (PDEs/PTEs): 75%

L1 hit rate (data): 90%

Derive a formula for and draw a graph of the access time for loads vs the TLB hit rate from 0-100%.



# TLB Hit Rates

$$EAT = P_{hit} T_{miss} + (1 - P_{hit}) T_{miss}$$

L1 cache hit: 1ns

L1 cache miss: 8ns

TLB cache hit: 0ns

L1 hit rate (PDEs/PTEs): 75%

L1 hit rate (data): 90%

Derive a formula for and draw a graph of the access time for loads vs the TLB hit rate from 0-100%.

$$EAT = (0 + (1 - P_{tlbhit})(T_{tlbmiss})) + (EAT_{l1})$$

$$T_{tlbmiss} = 2 * (0.75 * 1 + 0.25 * 8)$$

$$EAT_{l1} = 0.9 * 1 + 0.1 * 8$$