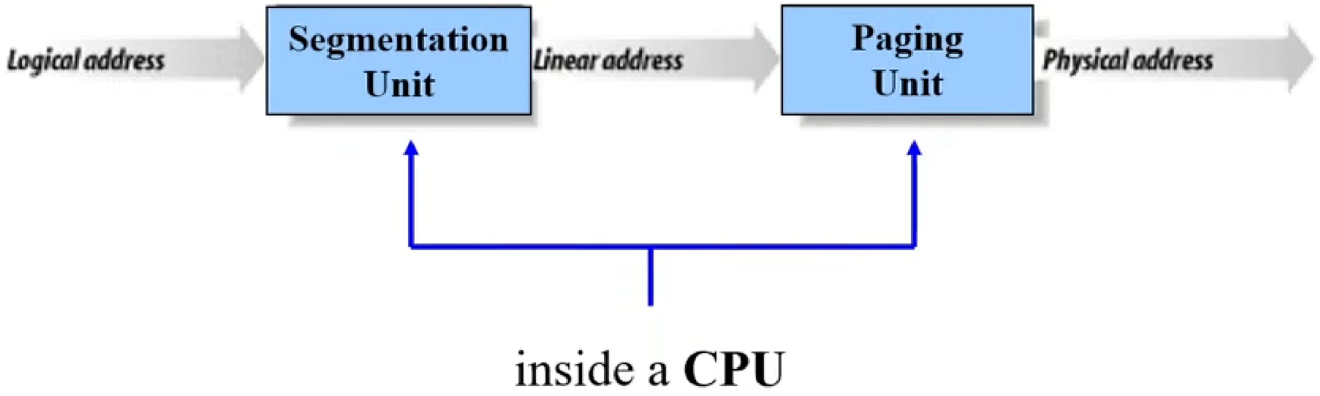
Memory-Addressing



{: width="2px"}

The Paging Unit

- A hardware circuit.
- Translates **linear addresses into physical ones**.
- Checks the requested access type against the access rights of the linear address.
    - If the memory access is not valid, it generates a **Page Fault** exception

Page Frame

- The paging unit thinks of all RAM as partitioned into **fixed-length page frames (physical pages)**.
- The size of a page is equal to the size of a page frame.
    - Usually the size of a page frame is 4KB; however, sometimes a larger page frame size may also be used.

Page

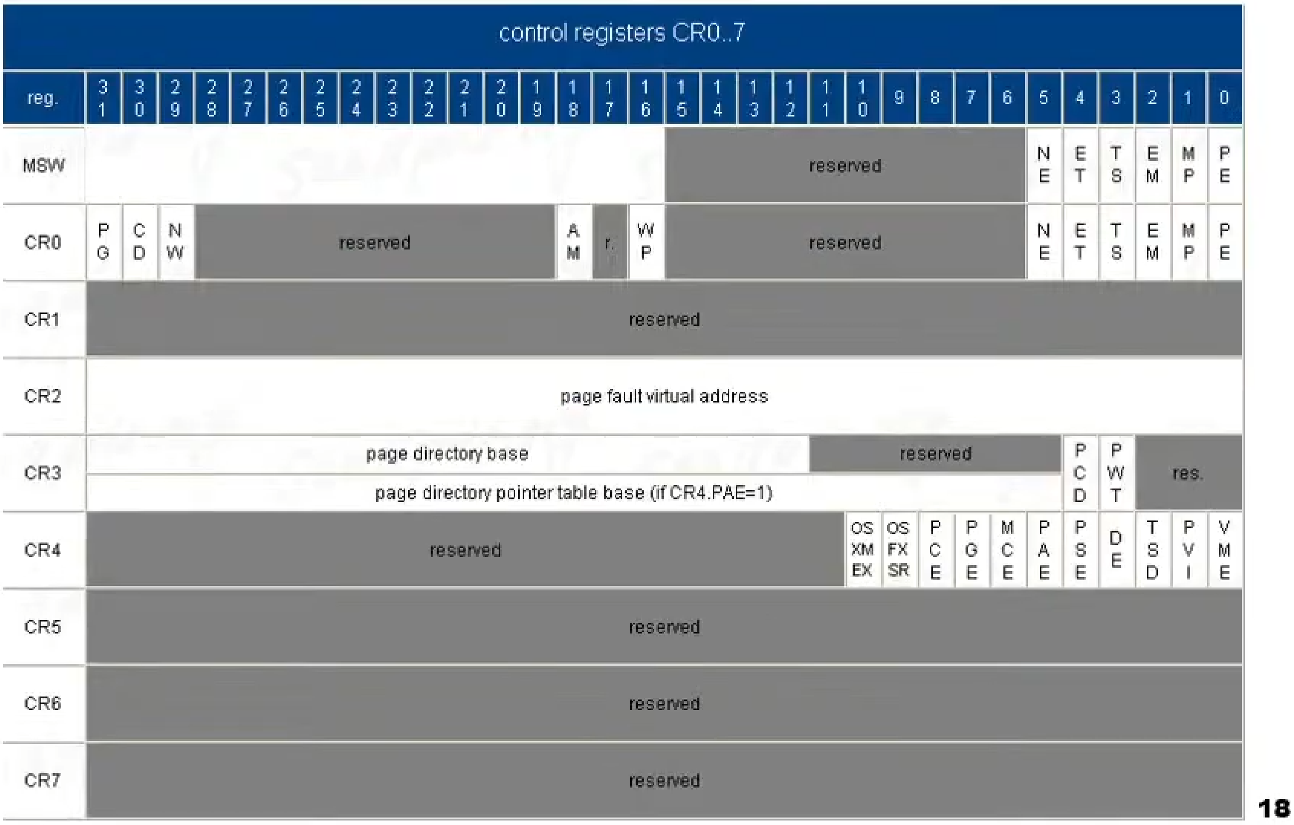
- Contiguous linear addresses are grouped in fixed-length intervals called pages.
- The term “page” is also refer to:
    - A set of linear addresses
    - The data contained in this group of addresses.

> Page 既可以指位址的範圍，也可以指在該範圍內存儲的數據

Enable Paging

- Starting with the 80386, all 80x86 processors support paging; paging is enabled by setting the PG flag of the control register cr0.
- When PG flag=0, **a virtual address is equal to a physical address**.
- Paging mechanism is used in protected mode.

Control Registers



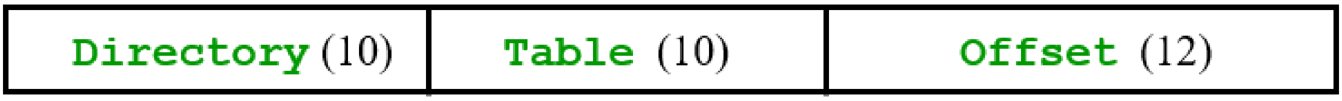
control registers CR0..7

reg.	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MSW																	reserved										NE	ET	TS	EM	MP	PE
CR0	PG	CD	NW	reserved										AM	r.	WP	reserved										NE	ET	TS	EM	MP	PE
CR1	reserved																															
CR2	page fault virtual address																															
CR3	page directory base																				reserved							PCD	PWT	res.		
	page directory pointer table base (if CR4.PAE=1)																															
CR4	reserved																					OSXMMEX	OSFXSR	PCE	PGE	MCE	PAE	PSE	DE	TSD	PVI	VME
CR5	reserved																															
CR6	reserved																															
CR7	reserved																															

18

Division of a Virtual Address

- A 32-bit virtual address is divided into 3 parts:
  - Directory: the 10 most significant bits.
  - Table: the 10 intermediate bits
  - Offset: the 12 least significant bits.



Directory (10)	Table (10)	Offset (12)

Translation Tables

- The translation of linear addresses is accomplished in two steps, each based on a type of translation tables.
- The first translation table is called the **Page Directory**, and the second is called the **Page Table**.
  - 小寫的「page table」指的是任何儲存線性位址和物理位址之間映射關係的頁。
  - 大寫的「Page Table」指的是最底層的頁表中的頁，用於最終的地址映射。
  - page table = Page Table OR Page Directory
- 每個 process 只有一個 page directory table.
- Each active process must have a Page Directory assigned to it.
  - The physical address of the Page Directory of the active process is stored in the control register cr3.
- Both of the above tables are located in main memory.

- Are initialized by kernel, before paging mechanism is activated.
- Allocating page frames to a **page table** occurs only when the process needs to access it. 記憶體的分配是按照需求

Paging of 80x86 -- The Directory Field

- The Directory field within the virtual address determines the entry in the Page Directory that points to the proper page table.
    - Hence, there are 2^10 entries in a Page Directory. 32-bit virtual address 前10個
    - Because each entry's size is 4 bytes; a Page Directory uses 4 * 2^10 = 4KB.

Paging of 80x86 -- The Table Field

- The address's Table field, in turn, determines the entry in the Page Table that contains the physical address of the page frame containing the page.
    - 同上有 2^10 entries, 所以 a Page Table uses 4 KB.

Paging of 80x86 -- The Offset Field

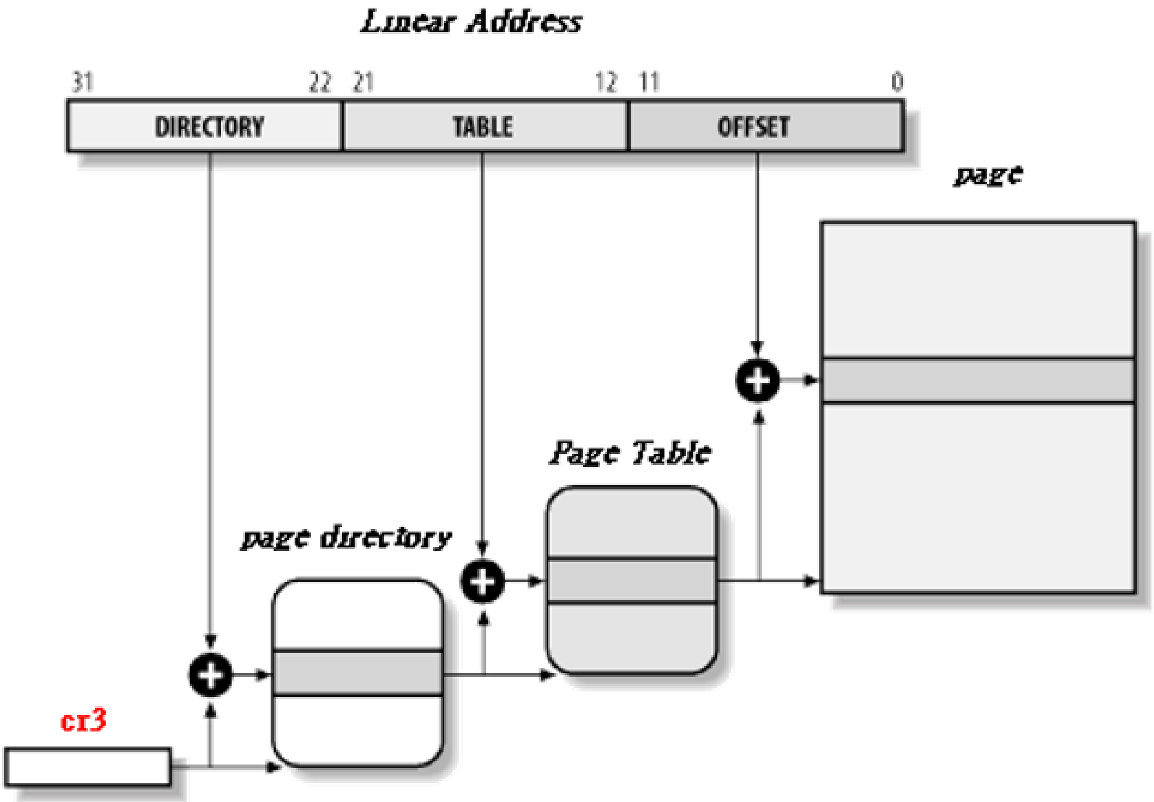
- The offset field determines the relative position within the page frame.
    - Each page frame consists of 4096 (i.e. 2^12) bytes of data.

> 2^32 = 4Gb => 所有地址共 4Gb -> 切 1 :3 給 kernel 跟 user

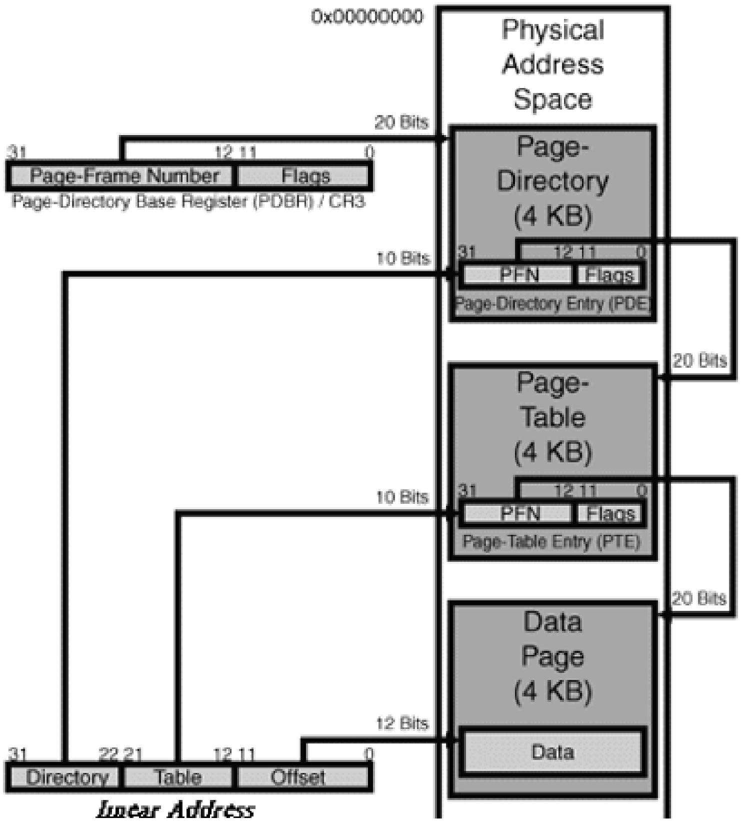
> 對 Directory 來說也是 1:3 -> 後面 256 給 kernal, 前面 768 給 user

> 相比於使用一層的(Only dir no table), 當第一層沒有使用到時, 就不用設定第二層, 可以節省空間

Paging by 80x86 Processors



Double-Layered Paging with 4-**KB** Pages



Why Use a Two-Level Scheme ?

- Reduce the amount of RAM required for per-process page tables.
  - Assume a process's maximum virtual address space is 4 GB.
    - For a single level scheme, 2^20 entries are needed.
    - If each translation table entry requires 4 bytes, then each process needs 2^20*4=4MB memory to store its translation table.

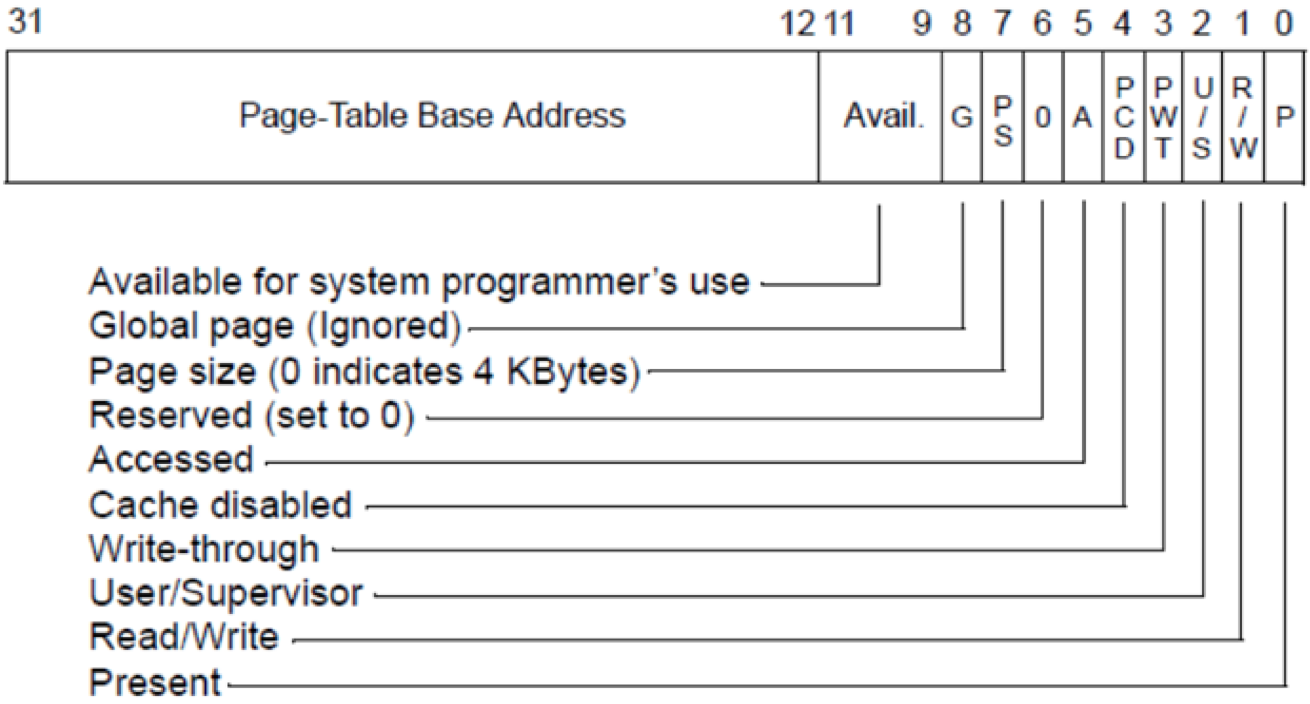
> 這樣的需求非常龐大，尤其是當系統中有許多進程時，會導致大量的記憶體被分頁表佔用。

- For a two-level scheme, translation tables are used only for those virtual memory regions actually used by a process.

> 按需分配：在兩級分頁中，只有當進程實際使用某些虛擬記憶體區域時，才會為該區域分配相應的頁表。未使用的虛擬地址空間則不需要頁表的映射，可以節省內存

P.S.: For most processes, most virtual memory regions are not used.

Structures of Page Directories And Page Tables Entries



由於 Dir 跟 Table 都只有 10 bit -> 其他部分(32-20=12)可以做別的用途

- Both Page Directory entries and Page Tables have the same structure.
  - Present flag
  - Field containing the 20 most significant bits of a page frame physical address.
  - Access flag
  - Dirty flag
  - Read/write flag
  - User/Supervisor flag
  - PCD and PWT flags

  - Page size flag
  - Global flag
- Present flag:
  - 1: yes
  - 0: no(page frame 不在物理記憶體中).
    - Save the virtual address -> cr2
    - Issue the Page Fault Exception.

> project 1-2 ?

- 20-bit physical address field:
  - Contain the 20 most significant bits of a page frame physical address.
  - The size of Page Directories, Page Tables, and page frame are all 4k bytes; therefore, the first physical address of the above entities is a multiple of 4 KB.
  - In other words, the physical address's least 12 significant bits are always zero and there is no need to store these 12 bits.

> 只有後面 20 位有值(前 10 位都是0,不需要儲存) -> 所以是 2^12=4KB 的倍數

- Accessed flag:
  - Set each time the paging unit addresses the corresponding page frame.
  - When swapping out a page frame is needed, OS uses this flag as a parameter to decide which page frame should be swapped out.

> 每當被訪問時，這個標誌會被設置，表示近期被使用過。

- Dirty flag.
  - **Apply to Page Table entries only**.
  - When a write operation is performed on a page frame, its corresponding Page Table entry's Dirty flag is set.
  - As the Accessed flag, this flag is also used by OS when determining choosing which page frame to swap out.

> 當有對 page frame 有寫入時, 要設置 dirty flag

- The paging unit never resets the above two flags; this must be done by the operating system.

> 找 Accessed flag 跟 Dirty flag 為 0 的, 把 Page Table 指向的 page 回收

- Read/Write flag:

  - Contain the access right (Read/Write or Read) of the page or the Page Table.

- User/Supervisor flag:

  - Contains the privilege level required to access the page or Page Table.

> 調整這個可以控制 user space 跟 kernel space

- PCD and PWT flags:
  - Controls the way the page or Page Table is handled by the hardware cache.
- Page Size flag:

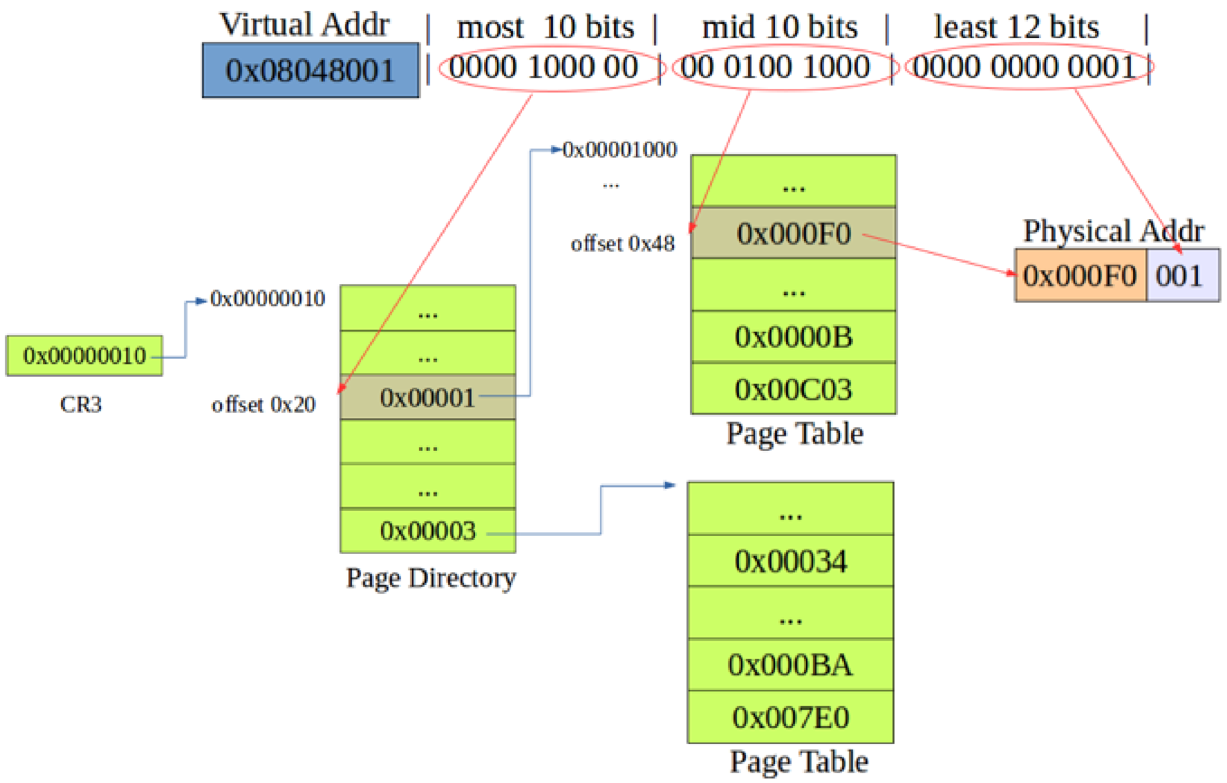
    - Apply only to Page Directory entries:
        - If it is set, the entry refers to a 2 MB– or 4 MB-long page frame.

> 跟後面的 Extended Paging 有關

- Global flag:
    - Applies to Page Table entries only to prevent frequently used pages from being flushed from the TLB cache.
    - Is used with the Page Global Enable (PGE) flag of cr4 register.

Example : x32

Example [REN]

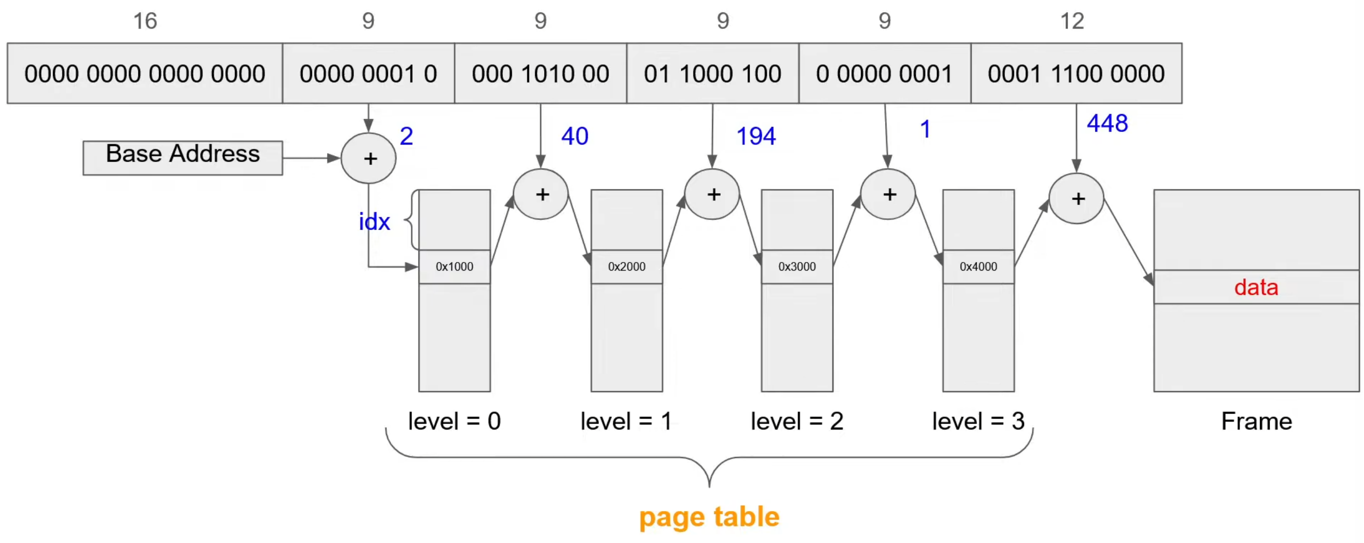


> 每個小格子是 4 bytes

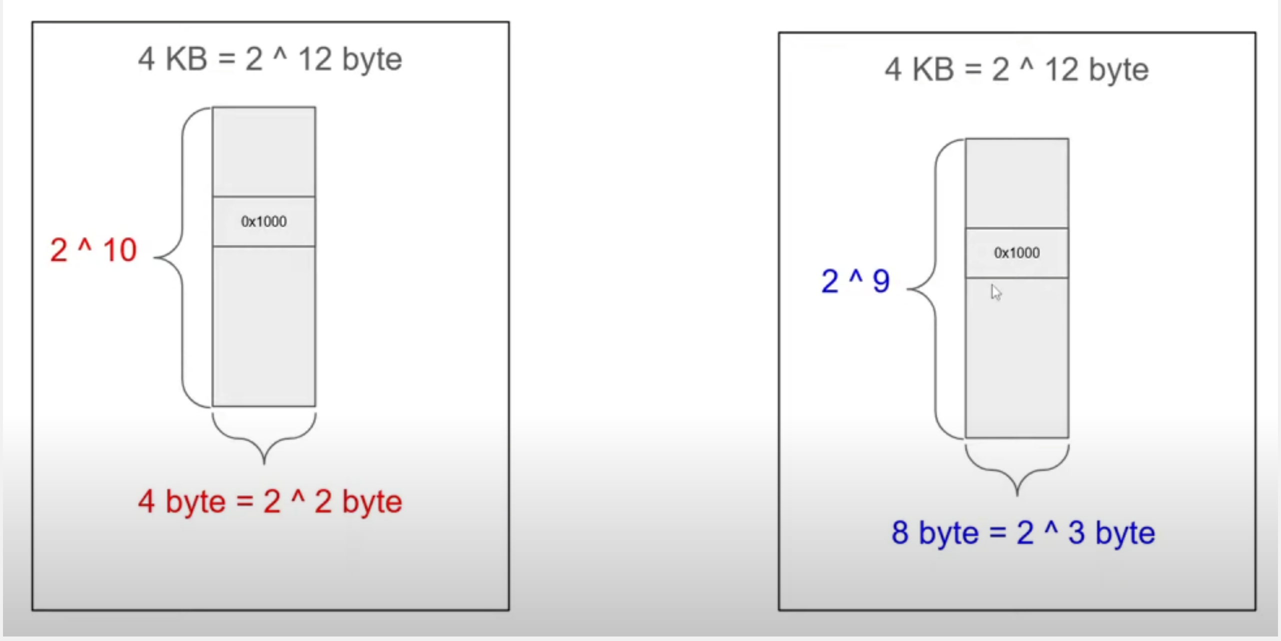
Example : x64

Page Table

[Translation fault]: 0x10a188011c0 = 1 0000 1010 0001 1000 1000 0000 0001 0001 1100 0000 (Virtual Address)



> 每個小格子是 8 bytes

> 為啥是 9 bits? 讓整個 table 維持 4KB
>
> 
>

Extended Paging

Why Extended Paging Is Introduced?

- Introduced starting from the Pentium model.
- Allows page frames to be 4 MB instead of 4 KB in size.
- Extended paging is used to translate large contiguous linear address ranges into corresponding physical ones.

> 可以將大範圍的連續線性位址直接轉換為對應的物理位址

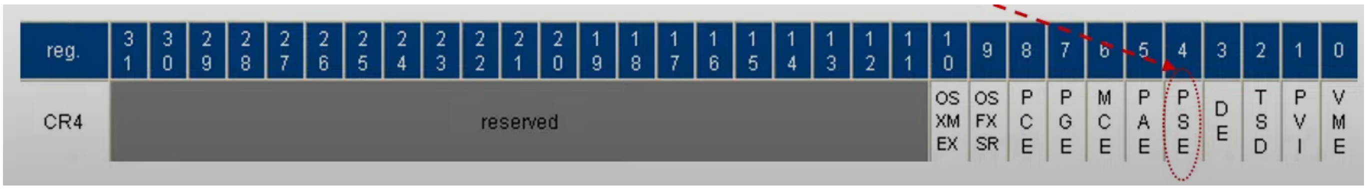
- In these cases, the kernel can do without intermediate Page Tables and thus save memory and preserve TLB entries.

> 不需要中間的頁表，從而節省記憶體，並保留更多的 TLB

Enable Extended Paging

- Is enabled by
  - setting the Page Size flag of a **Page Directory** entry.
  - setting the PSE flag of the cr4 processor register.

> 當 page directory 設定 PSE, 下一層就不是 table 而是 4MB 的 page



reg.	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CR4	reserved																					OS XM EX	OS FX SR	PCE	PGE	MCE	PAE	PSE	DE	TSD	PVI	VME

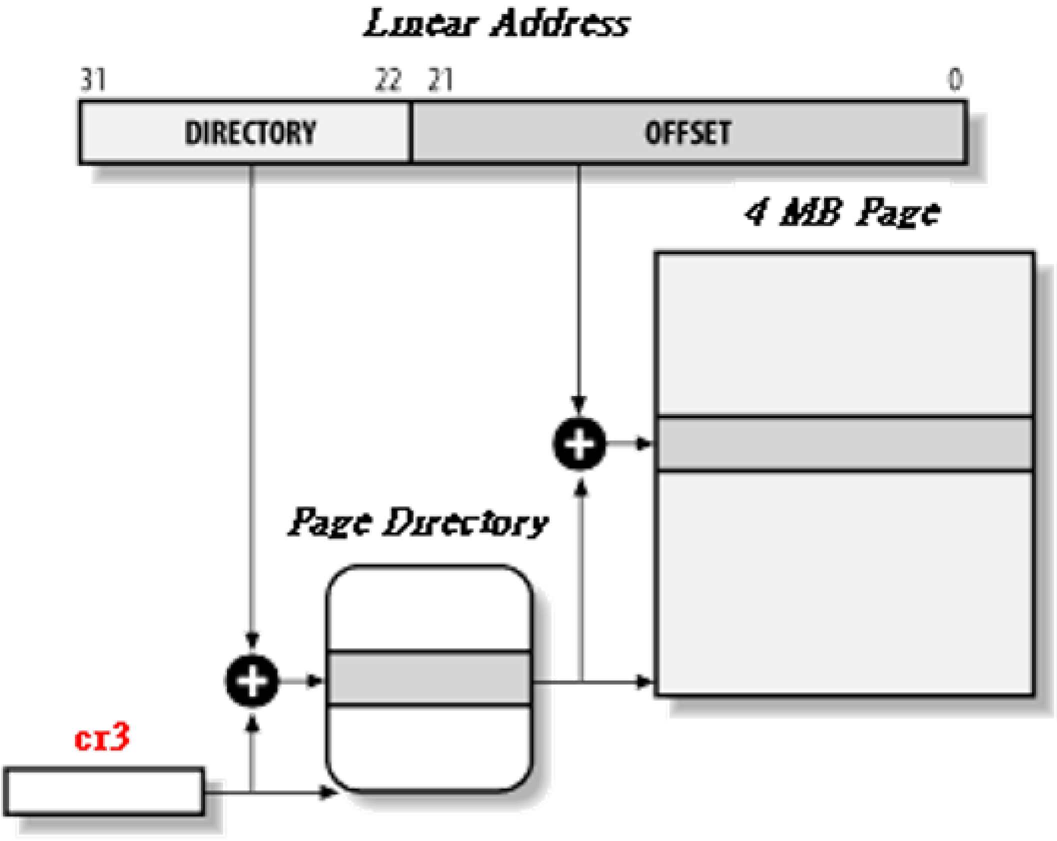
Virtual Address Layout under Extended Paging

- Under extended paging, the paging unit divides the 32 bits of a linear address into two fields:
  - Directory (10 bits).
  - Offset (22 bits; P.S.: 2^22=4MB)

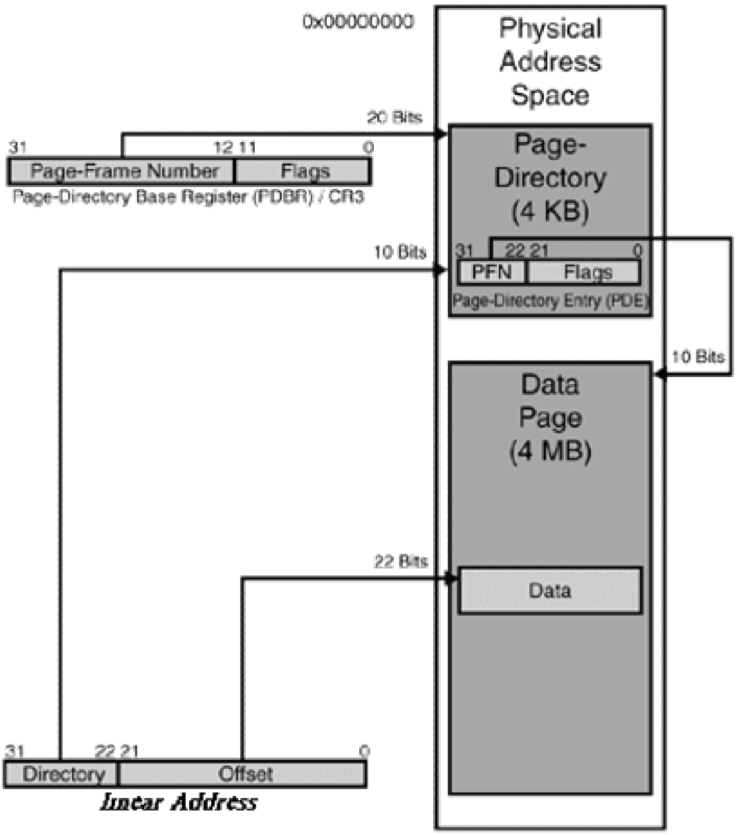
New Futures of Page Directory Entries under Extended Paging

- Under extended paging, the structure of a Page Directory and the entries inside it are the same as those in regular paging, except:
  - The Page Size flag is set.
  - Only the 10 most significant bits of the 20-bit physical address field are significant.

> 只有後10位有效(之前是20位)



Single-Layered Paging with 4-**MB** Pages



Hardware Protection Scheme

Privilege Levels

- The segmentation unit uses four possible privilege levels to protect a segment (the two-bit request privilege levels, 0 for kernel mode, 3 for user mode).
- The paging unit uses a different strategy to protect Page Tables and page frames □ the User/Supervisor flag.
  - 0 -> CPU's CPL must be less than 3 (i.e. for Linux, when the processor is in kernel mode.)
  - 1 -> the corresponding Page Table or page frame can always be accessed.

> paging unit 只有 1 bit

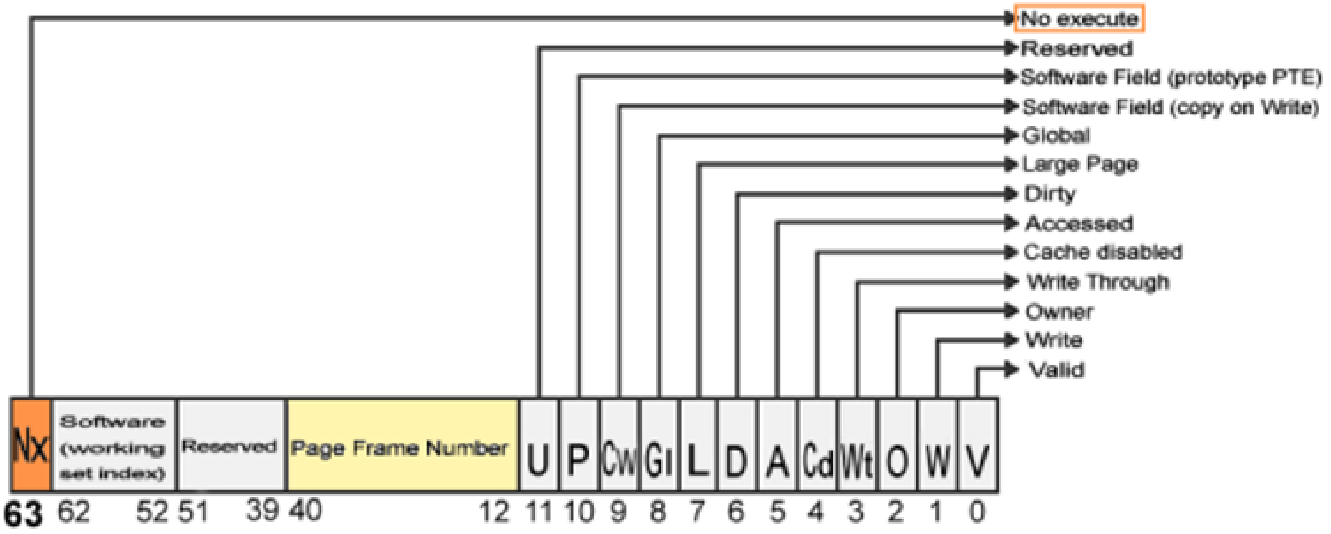
Access Rights

- Instead of the three types of access rights (Read, Write, Execute) associated with segments (determined by the type field of a segment descriptor), only two types of access rights (Read, Write) are associated with page tables and pages and are determined by the Read/Write flags of corresponding page tables entries.
  - Read/Write flag:
    - 0: can be read.
    - 1: can be read and write.

> paging unit 只有 1 bit

What's the "NX bit"?

- NX (No eXecute) bit actually refers, on x86 architectures, to the most significant bit (i.e. the 63th, or leftmost) of a 64- bit Page Table Entry.
- If this bit is set to 0, then code can be executed from that particular page.（可執行）
- If it's set to 1, then the page is assumed to only retain data, and code execution should be prevented. (僅用於儲存數據，不允許執行代碼)



Why Using NX?

- 在 80286 和其他較舊的架構中，內存劃分為多個 segments。然而，這樣的控制僅能在 segments 這一較大的單位上進行，而不能精確到頁或更小的記憶體單位。 -> 缺乏靈活性。

- x86 架構中的 NX 位元可以在頁級別（例如 4 KB 的頁框）上標記內存區域是否可執行，提供了更精細的內存保護。

The Physical Address Extension (PAE) Paging Mechanism

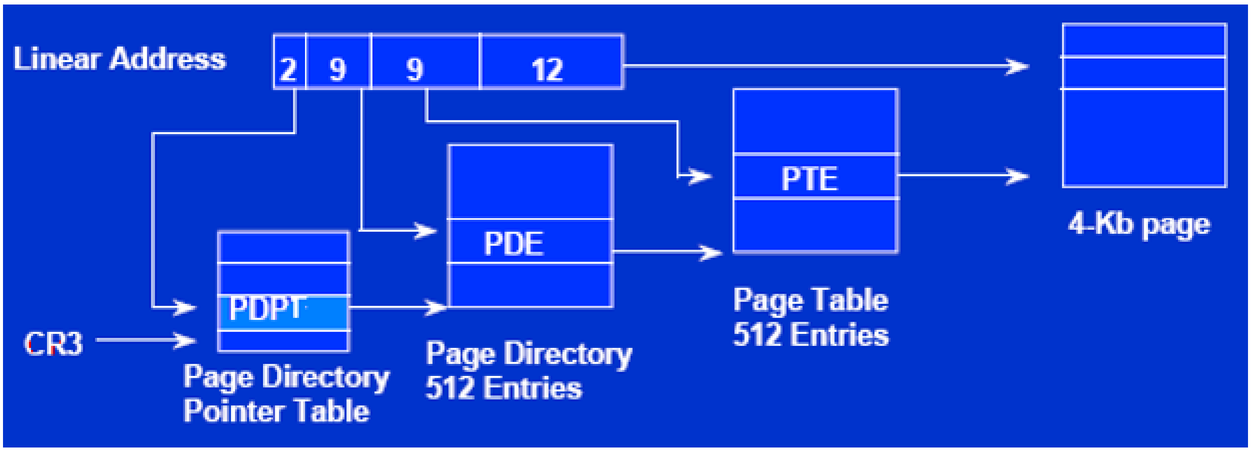
- Starting with the Pentium Pro, all Intel processors have 36 address lines; therefore, they are now able to address 2^36=64GB of RAM when is in PAE mode.
- PAE is activated by setting the Physical Address Extension (PAE) flag in the **cr4 control register**.
- Question: CPU registers such as EIP, ESP, are still 32 bits; thus, how to transfer a 32-bit virtual address into a 36-bit physical one?
- Answer: Introduce a new paging mechanism.
- The 64 GB (= 224x212) of RAM are split into 224 4-KB page frames.
- The entry size of Page Directories or Page Tables is increased from 4 bytes to 8 bytes; thus, each 4-KB page frame contains 512 (=2^9) entries instead of 1024 entries.
- The address field of each page table entry is increased form 20 bits to 24 bits; therefore, the address field can point to any of the 2^(36-12) => 2^24 4-KB page frames.

> 原本只有 2^20 4-KB page frames, 且記憶體最多 4 GB

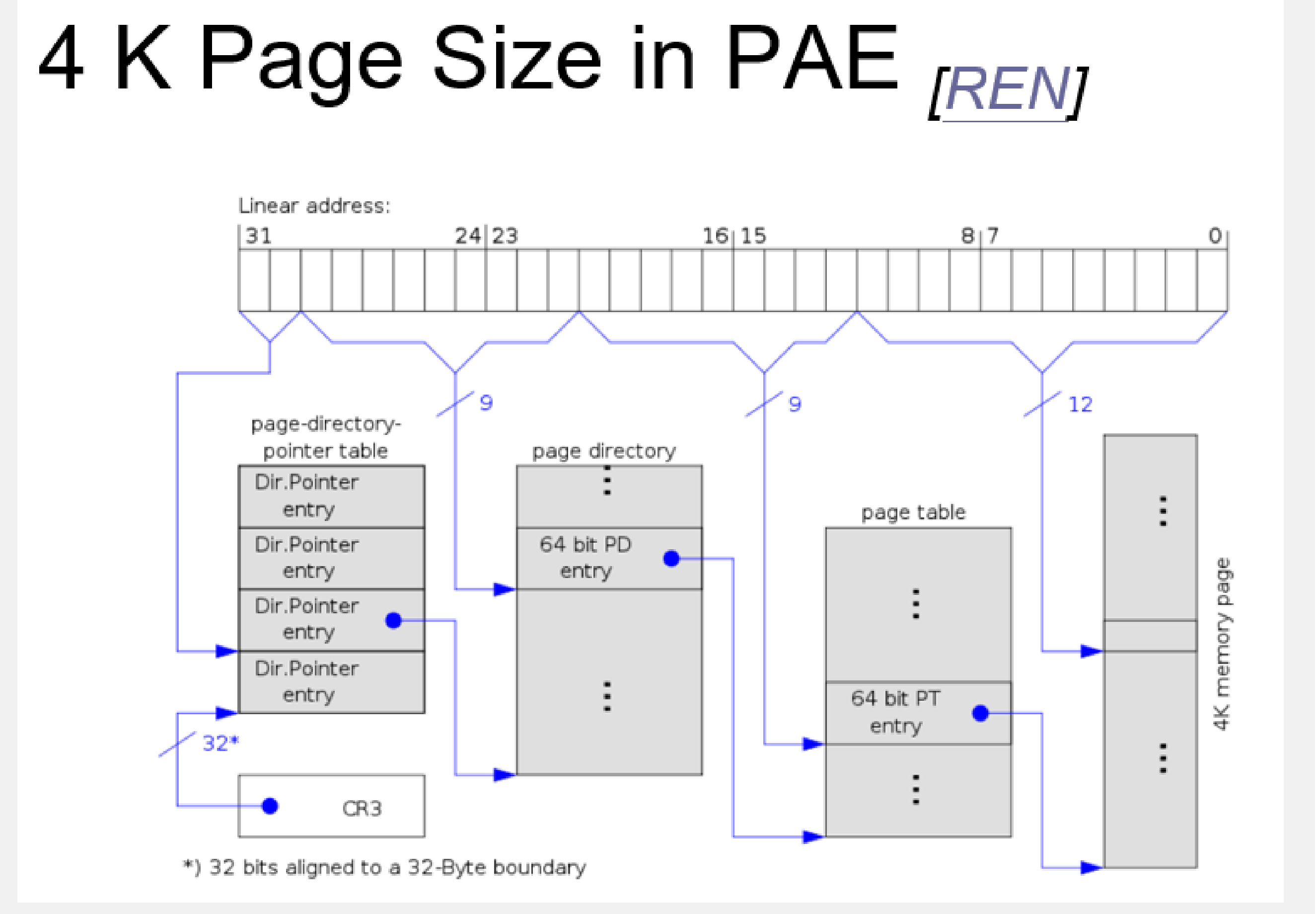
- A new level of page table is introduced --- **the Page Directory Pointer Table (PDPT)**

> 多了 4 條線, PAE 模式透過引入三層分頁機制 和 8 bytes 的頁表項，讓 32 位元的虛擬位址可以轉換成 36 位元的物理位址，從而支持更大的物理記憶體空間（最高 64 GB）

- When PAE is activated, and the PS flag in Page Directory is cleared (i.e. each page frame is 4KB), a virtual address is split into the following four fields PDPT(2 bits), PD(9 bits), PT(9 bits), Offset(12 bits).



> PDPT 只能放在 64GB 中的前 4GB 因為這樣前四個值為 0, CR3 (32bits) 左邊直接加四個0 所以還要是 32 的倍數 => 右邊五個為0 這樣 CR3 需要 27 個值 (4+27+5=36)



4 K Page Size in PAE [REN]

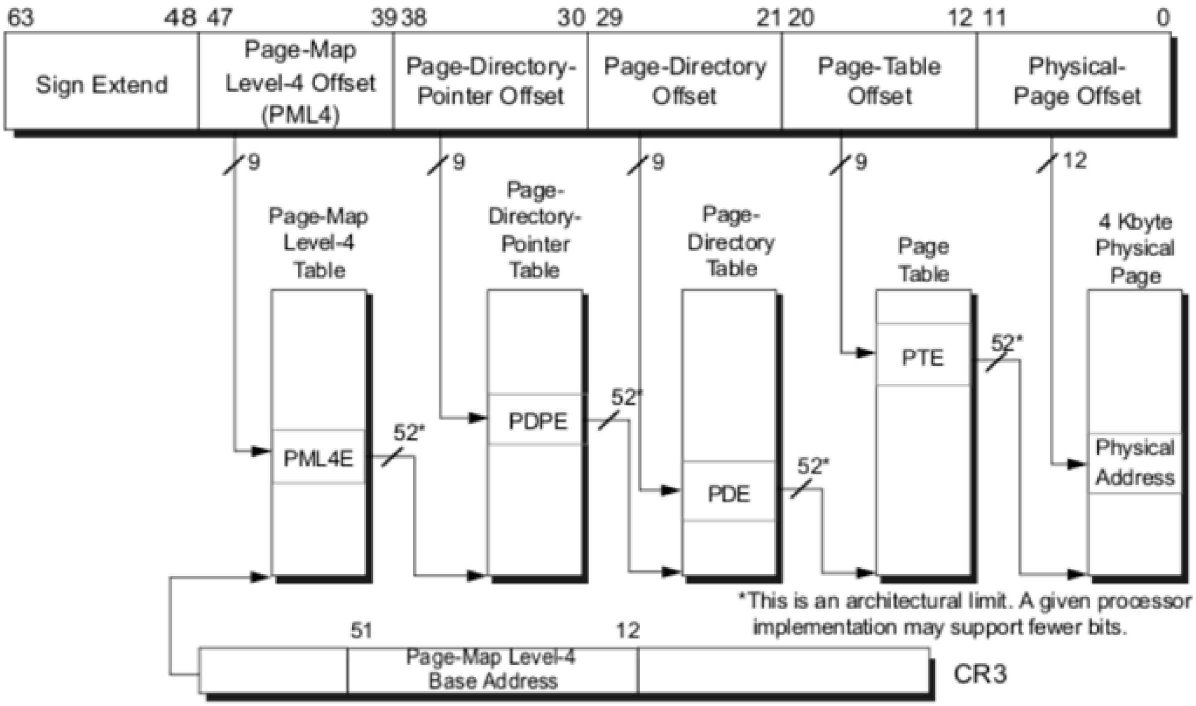
> 傳統的 32 位系統的限制： 在沒有 PAE 機制的情況下，無論系統有多少 RAM，處理器最多只能訪問前 4 GB 的 RAM，因為傳統的 32 位位址空間只能表示 4 GB 的位址範圍。

> PAE 機制的改進： 在啟用了 PAE 的情況下，即使系統具有 64 GB 的 RAM，系統可以隨意訪問 64 GB RAM 中的任何頁框子集。這個子集的大小依然限制為 4 GB，但可以在更大的物理地址範圍中動態映射所需的頁框。

Paging for 64-bit Architectures

- All hardware paging systems for 64-bit processors make use of additional paging levels. The number of levels used depends on the type of processor.

64-bit Four-Level page table Hierarchy [REN]



> 只有用到 48 bits

Locality Types

- Temporal locality
  - The concept that a resource that is referenced at one point in time will be referenced again sometime in the near future.

> 時間區域性指的是在某個時間點被訪問的資源（如記憶體位置、資料）在不久的將來很可能會再次被訪問。 Ex. 循環變量會被多次使用，因此循環變量的存取具有高度的時間區域性

- Spatial locality
  - The concept that likelihood of referencing a resource is higher if a resource near it was just referenced.

> 空間區域性指的是如果一個資源（例如一個記憶體位置）剛被訪問，那麼其相鄰的資源被訪問的可能性也較高。 Ex.陣列存取

- Sequential locality
  - The concept that memory is accessed sequentially.

> 當程式按順序讀取一個檔案的內容，或遍歷一個陣列時，這些操作符合順序區域性