分頁架構

設定分頁功能

在控制暫存器（control registers）中，有三個和分頁功能有關的旗標：PG（CR0 的 bit 31）、PSE（CR4 的 bit 4，在 Pentium 和以後的處理器才有）、和 PAE（CR4 的 bit 5，在 Pentium Pro 和 Pentium II 以後的處理器才有）。PG（paging）旗標設為 1 時，就會開啟分頁功能。PSE（page size extensions）旗標設為 1 時，才可以使用 4MB 的分頁大小（否則就只能使用 4KB 的分頁大小）。而 PAE（physical address extension）是 P6 家族新增的功能，可以支援到 64GB 的實體記憶體（本文中不說明這個功能）。

分頁目錄和分頁表

分頁目錄和分頁表存放分頁的資訊（參考「記憶體管理」的「緒論」）。分頁目錄的基底位址是存放在 CR3（或稱 PDBR，page directory base register），存放的是實體位址。在開啟分頁功能之前，一定要先設定好這個暫存器的值。在開啟分頁功能之後，可以用 MOV 命令來改變 PDBR 的值，而在工作切換（task switch）時，也可能會載入新的 PDBR 值。也就是說，每一個工作（task）可以有自己的分頁目錄。在工作切換時，前一個工作的分頁目錄可能會被 swap 到硬碟中，而不再存在實體記憶體中。不過，在工作被切換回來之前，一定要使該工作的分頁目錄存放在實體記憶體中，而且在工作切換之前，分頁目錄都必須一直在實體記憶體中。
分頁目錄和分頁表的 entry 格式如下：

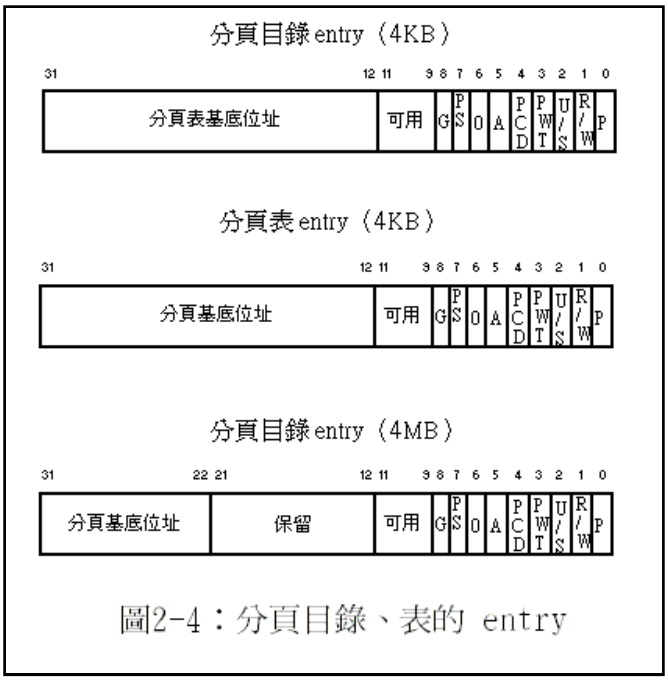

圖2-4：分頁目錄、表的 entry

上表中的各個欄位的說明如下：

- 分頁表基底位址、分頁基底位址：存放分頁表 / 分頁的基底位址（以實體位址）。在 4KB 的分頁中，分頁表和分頁的位址都必須是 4KB 的倍數，所以用 20 bits 來表示基底位址的最左邊的（most-significant）20 bits。在 4MB 的分頁中，分頁的位址必須是 4MB 的倍數，因此用 10 bits 表示基底位址最左數的 10 bits。
- P（present）旗標：表示這個分頁（或分頁表）目前是否存在記憶體中。若 P = 1，則表示這個分頁或分頁表在記憶體中，可以進行位址轉換。若 P = 0，則表示這個分頁不在記憶體中，若對這個分頁進行存取動作，會導致 page fault（#PF）例外。作業系統在將分頁 swap 到硬碟時，要把 P 設為 0；而在把分頁由硬碟中讀入時，則要把 P 設為 1。
- R/W（read/write）旗標：當 R/W = 1 時，表示分頁可以寫入；當 R/W = 0 時，表示分頁只能讀取（read-only）。當 CR0 的 WP 旗標（第 16 bit）設為 1 時，所有的程式都不能寫入唯讀的分頁；但 WP 為 0 時，具有 supervisor 等級的程序就可以寫入唯讀的分頁。在指向分頁表的分頁目錄 entry 中，這個旗標對其指向的分頁表中的每個分頁都有效。
- U/S（user/supervisor）旗標：當 U/S = 1 時，表示分頁是一個 user level 的分頁，而 U/S = 0 時，表示分頁是一個 supervisor level 的分頁。和 R/W 旗標一樣，在分頁目錄中，這個旗標對其指向的分頁表中的每個分頁都有效。

- PWT（page-level write-through）旗標：在 PWT = 1 時，處理器會對這個分頁（或分頁表）做 write-through caching；而 PWT = 0 時，處理器會對這個分頁（或分頁表）做 write-back caching。在 CR0 的 CD（cache disable，第 30 bit）設為 1 時，這個旗標會被忽略。
- PCD（page cache disable）旗標：在 PCD = 1 時，處理器不會對這個分頁（或分頁表）進行 cache；而 PCD = 0 時，則會進行 cache。例如，在分頁是對映一 I/O 記憶體時，就需要把 cache 關閉。在 CR0 的 CD 旗標設為 1 時，這個旗標會被忽略。
- A（accessed）旗標：在 A = 0 時，若分頁被存取，則處理器會把它設為 1。在被設為 1 之後，處理器不會自動把它設為 0，只有軟體可以把它清為 0。因此，通常在一個分頁被載入實體記憶體時，作業系統會把 A 清為 0。記憶體管理程式或作業系統可以利用這個旗標來決定 swap 的方式。
- D（dirty）旗標：在 D = 0 時，若對分頁進行寫入動作，則處理器會把它設為 1。在被設為 1 之後，只有軟體可以把它清為 0。通常作業系統在載入一個分頁之後，會把 D 清為 0。如此一來，要把這個分頁 swap 到硬碟中時，若 D 仍為 0，則表示分頁沒有被修改過，就不需要再寫回硬碟中了。這個旗標在「指向分頁表的分頁目錄 entry」中沒有作用。
- PS（page size）旗標：這個旗標只在分頁目錄 entry 中有作用。當 PS = 0 時，表示這是一個 4KB 的分頁，因此 entry 是指向一個分頁表；當 PS = 1 時，表示這是一個 4MB 的分頁，因此 entry 是指向一個分頁。只有在 CR4 的 PSE（page size extensions，第 4 bit）為 1 時，才能存取 4MB 的分頁。
- G（global）旗標：這是在 Pentium Pro 及之後的處理器才有的旗標。在本文中不討論。在 Pentium 和之前的處理器，這個旗標視為保留旗標，必須設為 0。
- 保留和可用部分：保留部分一律要設成 0，而可用部分則可以自己決定用途。如果 P 為 0，則整個 entry（除了 P 之外）都視為可用部分，可供作業系統存放相關的資訊（例如，可以用來存放分頁在硬碟 swap file 中的位置）。

Translation Lookaside Buffers（TLBs）

到記憶體中查分頁目錄和分頁表是非常耗時的工作（需要經由較慢的 memory-bus），而查分頁目錄和分頁表又是非常頻繁的事件（幾乎所有的記憶體存取動作都需要），因此，處理器把最近使用的分頁目錄和分頁表的 entry 存放在叫 Translation Lookaside Buffers（TLBs）的 cache 中。只有 CPL 為 0 的程序才能選擇 TLB 的 entry 或是把 TLB 設為無效。無論是在更動分頁目錄或分頁表之後，都要立刻把相對的 TLB entry 設為無效，這樣在下次取用這個分頁目錄或分頁表時，才會更新 TLB 的內容（否則就可能會從 TLB 中讀到舊的資料了）。
要把 TLB 設為無效，只要重新載入 CR3 就可以了。要重新載入 CR3，可以用 MOV 指令（例如：MOV CR3, EAX），或是在工作切換時，處理器也會重新載入 CR3 的值。此外，INVLPG 指令可以把某個特定的 TLB entry 設成無效。不過，在某些狀況下，它會把一些 TLB entries 甚至整個 TLB 都設為無效。INVLPG 的參數是該分頁的位址，處理器會把 TLB 中存放該分頁的 entry 設為無效。

分頁的規劃

分頁機制在多工作業系統中是很重要的。在多工作業系統中，往往同時執行很多個程式，因此，記憶體可能常常會用盡。但是，即使一個程式載入大量的資料到記憶體中，也很少會同時使用到全部的資料。這時候，把暫時不需要的資料寫入硬碟（或其它類似的裝置）中，就可以空出位置載入其它的程式了。不過，為了管理的方便，分頁的大小往往是固定的。例如，在 IA-32 架構下，分頁的大小是 4KB。分頁如果太大，則在 swap 時，常常會 swap 到不需要 swap 的部分；而若分頁太小，則過於破碎，不易管理，也缺乏效率。
在 i486 和之前的處理器中，分頁的大小只有一種選擇：4KB。在大部分情形中，這個大小還算適當。但是，在某些情形下，可能會需要更大的分頁。因此，在 Pentium 和以後的處理器，就增加了 4MB 的分頁大小。然而，4MB 在一般的情形中，實在是太大了，實用性也降低。不過，4MB 的分頁在某些狀況下還是有用的。例如：為了方便管理，可以把作業系統的核心放在 4MB 的分頁中，而一般應用程式則使用 4KB 的分頁。此外，在 Linux 作業系統中還有一種用法：Linux 作業系統的核心部分常常需要使用實體位址，因此在 Linux 中，應用程式和核心是使用不同的分頁目錄。核心的分頁目錄便是將線性記憶體直接對映到實體記憶體中。在這種情形下，就很適合使用 4MB 的分頁模式。不過，要注意一點：4MB 的分頁模式，只有在 Pentium 及以後的處理器才能使用。