Virtual Memory

Virtual Memory: Virtual memory concept is used to map physical memory for processor. Virtual memory part resides in main memory and is used to access physical pages efficiently.

Let’s have a 32 bit Linux memory map:

- Virtual memory is a 32 bit memory location which has 20 bits reserved for page number and 12 bits for offset. so total number of possible page references can be $2^{12} = 4K$.
- How to perform this mapping ??????
  1. make a page table in memory:

One approach may be to map all physical pages in virtual page table. So there can be at most $2^{20}$ page entries and if we consider each page entry to be 4 K in size then total table size would be 4 MB. This approach takes large amount of memory so it is not a suitable approach.
2. **Two level hierarchical structure:**

In another scheme we split 20 bit page number entry into 2 parts, each of 10 bit in size. First part can be used for referencing page directory and other 10 bits can be used to find exact page number using data of page directory.
This kind of structure is an example of 4096-ary tries data structure.

Ex: Given address is 084075E0

so first 10 bits are: 0000 1000 01 = 33 (in base 10)

next 10 bits are: 0000 0001 11 = 7 (in base 10)

It's entry 33, which is the 34th entry, as entry 0 is the first entry.
In this approach as shown first 10 bits are used to resolve page directory entry and other 10 bits are used for resolving page table entry.

There are 3 memory accesses as follows:

00000084

PD[33] 00001

0000100C

PT[7] 00002

000025E0

some value

• **Translation look aside Buffer (TLB):** It is a cache for mapping page directory and page table scheme almost double the access time as it has 2 looks ups.

• TLB can make things faster. Every page is first searched in TLB, if it is found in that then control goes to physical page otherwise whole memory table is accessed as previous scheme.

• There are some issue with TLB scheme:
  o Performance: Consider a scenario where TLB has 384 entries and each entry has 4 K size then total size will be:
    \[384 \times 4 \text{ K} = 1.5 \text{ MB}\]
  o Correctness: There are some mapping which exist in TLB and 2 level tables also. For allocation or de-allocation of pages both the entries must be flushed but some time inconsistency occurs.