## vax-11 hardware

basic entity in VAX-11: process, byte addressable, 32-bit address space, containing 21-bit pages and each page contains 512 bytes.

basic unit of mapping and protection: page

\tip:

If the kernel is mapped in every single user process, so that there will not be TLB flush when the system call because now the kernel and user are in the same address space; protection is done by hardware ring.

In total four regions in the virtual space

P0(program) region - process unique - heap - growing to higher address

P1(control) region - process unique - stack - growing to lower address

system region - shared by all processes

reserved region

Each region is defined by a page table, which is an array of 32-bit page-table entries.

a valid bit - a protection field - modify bit - field for the OS - physical page frame number

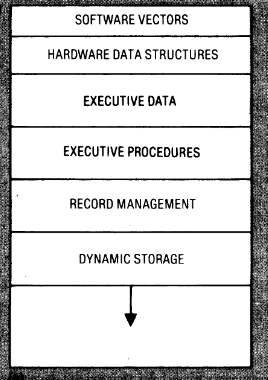
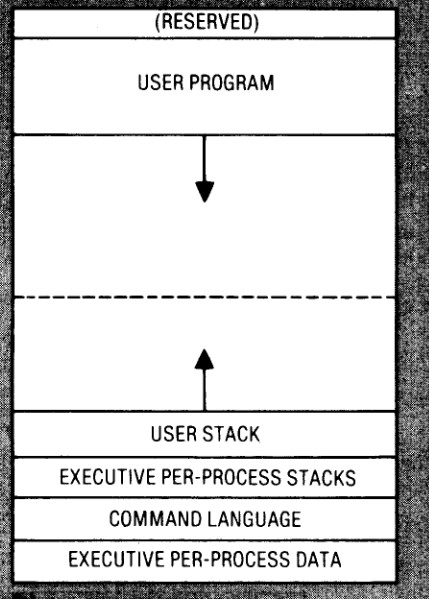
Each page table is defined by two hardware registers.

* base address register
* length register

a virtual-to-physical translation buffer is provided.

usage on different regions

system region

[[1]](#footnote-0)

executive code and data

process-specific data structures

process page tables

program region

user’s executable program

control region

process-specific data

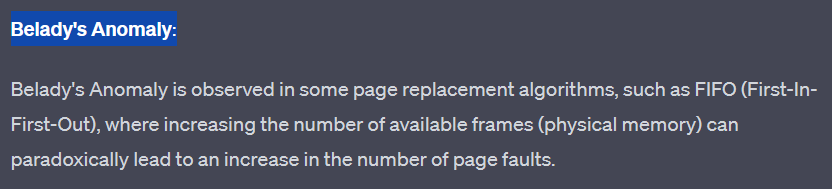
## memory management implementation

The memory management system is deviedde into two componrnet:

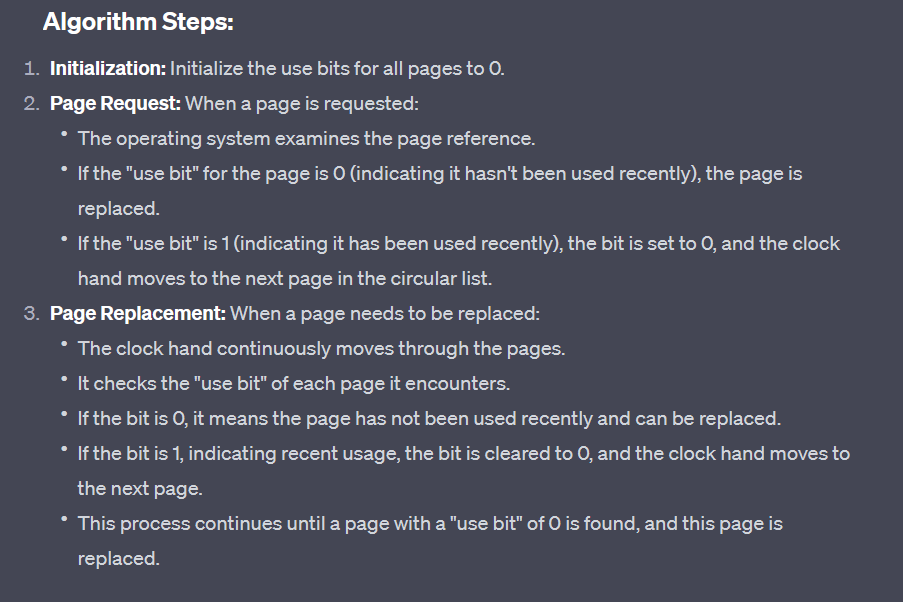
* pager: process the page fault and is responsible for removing and loading process pages into the main memory.
* swapper: process the page removing and loading for an entire process

the paging policy in VAX: replace the page from the process requesting for a new page. In the page set for one process, FIFO is used. [To measure the age of one page, a reference-bit on each page table entry is used.] —> Q: why there is need for measuring the age? since FIFO list is used? Why the FIFO is used here? just to reduce the reference cost?

\tip:FIFO is easy to implement, but is has problem of:



Similar to CLOCK(second-change) algorithm:



To reduce the cost of page fault -> software cache on the to-be-removed pages.

two page lists(different by the modify bit):

* free page list
* modified page list(act as a cache as well)

If there is page which will be quickly referenced to be remove, since the cost by returning a page from the two lists are minimal, the cost will be minimal. And if the page is returned from the free list, it’ll be on the FIFO list top.

To reduce the cost of paging I/O - clustering paging.

demand-zero and copy-on-reference are used.

The user process can have some control over its memory through a set of routings: expand, increase, lock the pages, lock the process, create, produce a record.

1. system vectors contain pointers to all the executive service routines, thus the location of these services can be changed. Also, the protection is done by the hardware mode protection. [↑](#footnote-ref-0)