Memory Management

Virtual memory: Each process has 2^{64} bytes of memory.

A page (block) of virtual addresses is somewhere in ram.

Process Page table: Virtual to Real
22000H → 3000H
per process.

Swap space: used as slow ram, some pages may be found here.
Page table indicates disk and location.
Kernel: does not get paged.

Buffers: I/O is buffered. Writes are delayed,
Prefetches are allowed.
if page is in a buffer when it requested
the copy in the buffer is used

Behavior:
`cc` delay while compiler is read from disk
`cc` no delay, compiler is still in RAM

LRU: least recently used.
Enough buffer: frequently used pages in RAM
Too little buffer: frequent disk accesses

`free`: memory summary

If your free memory or buffers go to small,
you lose performance due to excess disk activity

`ps`:
`VSZ`: virtual memory size: text (code) + data + stack
`RSS`: resident set size
Kilobytes of program in memory

`lpd`— `VSZ 31, RSS 0`
Program alive, using 31K of data + code
neither the program nor its data are in RAM.
Conclude: program idle.

`csh`— `VSZ 172, RSS 688`
Program alive, 172K
Active, in RAM, substantial memalloc.
using over 0.6MB real RAM

An idle process uses no RAM (swap space yes).
Conclusion: processes that are inactive don't hurt performance.
Conclusion: avoid servers that are active even when they aren't being used.

Swap Space

Actually: Paging space.
The area on disk where pages of virtual memory are stored when there is not enough real RAM.

Out of Core: Your requests exceeded the amount of swap space

Two types of swap space:
1) Swap partition (preferred)
2) Swap file (easy to add)

Swap partition

a) set up your partitions using fdisk.
label the partition as “swap” (82)

b) set up the swap structures in the partition

```
mkswap /dev/sdc1
```

lots of options, usually the defaults work

c) turn swapping on `swapon /dev/sdc1`

On shutdown turn swapping off `swapoff /dev/sdc1`

Swap file

a) create a contiguous file on disk

```
dd if=/dev/zero of=/myswap bs=1024 count=1024
```

1MB swapfile called `myswap`

b) set up the structures in the file

```
mkswap /myswap
```

c) turn swapping on

```
swapon /myswap
```

`/etc/fstab` can name swap partitions and files

```
swapon -a
```

turn on swapping for every swap thing listed in `fstab`

```
/dev/sdc1  none swap sw 0 0
/myswap    none swap sw 0 0
```

I/O Buffers

Amount of RAM is set by system (varies)

Behavior:

Write back: block will be written to disk when convenient.

Prefetch: read next block in file too.

Requirement: control writting buffers to disk
`bdflush`: does the write back
`update`: starts `bdflush`

Requirement: write before removing disk or shutting down

`sync`: write the buffers to disk.

`umount`: forces a `sync` for that disk.

Note: many drives can be “locked” by software.
Linux: lock drives with mounted file systems or active swap space.