Swap

coursera.org/learn/linux-for-developers/supplement/YDzQB/swap

Linux employs a virtual memory system in which the operating system can function as if it had more memory than it really does. This kind of memory overcommission functions in two ways:

- Many programs do not actually use all the memory they are given permission to use. Sometimes, this is because child processes inherit a copy of the parent's memory regions utilizing a COW (Copy On Write) technique, in which the child only obtains a unique copy (on a page-by-page basis) when there is a change.
- When memory pressure becomes important, less active memory regions may be swapped out to disk, to be recalled only when needed again.

Such swapping is usually done to one or more dedicated partitions or files; Linux permits multiple swap areas, so the needs can be adjusted dynamically. Each area has a priority and lower priority areas are not used until higher priority areas are filled.

In most situations, the recommended swap size is the total RAM on the system. You can see what your system is currently using for swap areas with:

```
1
2
3
4
$ cat /proc/swaps
Filename                Type            Size        Used        Priority
/dev/sda9               partition       4193776     0           -1
/dev/sdb6               partition       4642052     0           -2
```

and current usage with:

```
1
2
3
```

```
4

$ free
             total       used       free     shared    buffers     cached
Mem:       4047236    3195080     852156          0     818480    1430940
Swap:      8835828          0    8835828
```

The only commands involving swap are **mkswap** for formatting a swap file or partition, **swapon** for enabling one (or all) swap area, and **swapoff** for disabling one (or all) swap area.

At any given time, most memory is in use for caching file contents to prevent actually going to the disk any more than necessary, or in a sub-optimal order or timing. Such pages of memory are never swapped out as the backing store is the files themselves, so writing out to swap would be pointless; instead, dirty pages (memory containing updated file contents that no longer reflect the stored data) are flushed out to disk.

It is also worth pointing out that Linux memory used by the kernel itself, as opposed to application memory, is never swapped out, in distinction to some other operating systems.