

Resumen Adress Spaces - SistOp

Lautaro Bachmann

Contents

The Abstraction: Address Spaces	3
The Address Space	3
Definition	3
The memory state elements	3
Parts that may grow (and shrink)	3
the abstraction	3
Goals	3
transparency.	3
efficiency.	4
protection.	4
Summary	4
The VM system	4
The OS,	4

The Abstraction: Address Spaces

The Address Space

Definition

it is the running program's view of memory in the system. The address space of a process contains all of the memory state of the running program.

The memory state elements

the code

The code of the program (the instructions) have to live in memory somewhere,

stack

The program uses a stack to keep track of where it is in the function call chain as well as to allocate local variables and pass parameters and return values to and from routines.

the heap

is used for dynamically-allocated, user-managed memory,

there are other things

There are other things (e.g., statically-initialized variables), but for now let us just assume those three components: **code, stack, and heap.**

Parts that may grow (and shrink)

The parts that may grow (and shrink) are the heap (at the top) and the stack (at the bottom).

We place them like this because each wishes to be able to grow, and by putting them at opposite ends of the address space, we can allow such growth: they just have to **grow in opposite directions.**

this placement of stack and heap is just a convention;

the abstraction

The program really isn't in memory at physical addresses 0 through 16KB; rather it is loaded at some arbitrary physical address(es).

Goals

transparency.

The OS should implement virtual memory in a way that is invisible to the running program. the program behaves as if it has its own private physical memory.

efficiency.

The OS should strive to make the virtualization as efficient as possible, both in terms of time and space

protection.

The OS should make sure to **protect** processes from one another as well as the OS itself from processes.

Summary**The VM system**

is responsible for providing the illusion of a large, sparse, private address space to programs, which hold all of their instructions and data therein.

The OS,

The OS, with some serious hardware help, will take each of these virtual memory references, and turn them into physical addresses, which can be presented to the physical memory in order to fetch the desired information.

The OS will do this for many processes at once, making sure to protect programs from one another, as well as protect the OS.