Extended OS

Learning Outcomes

- An appreciation that the abstract interface to the system can be at different levels.
  - Virtual machine monitors (VMMs) provide a low-level interface
- An understanding of trap and emulate
- Knowledge of the difference between type 1 and type 2 VMMs
- An appreciation of some of the issues in virtualising the R3000

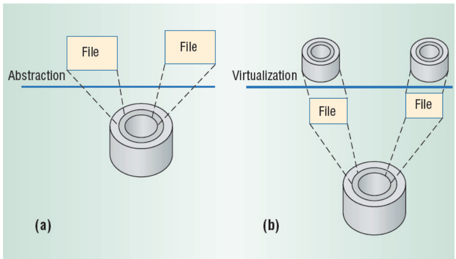
Virtual Machines

References:

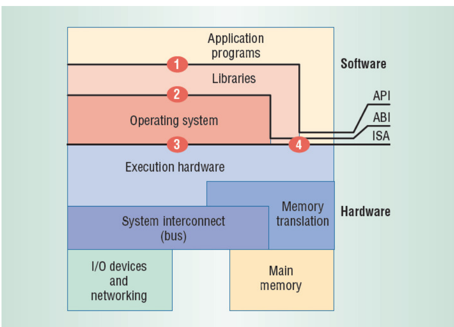
Smith, J.E.; Ravi Nair; , "The architecture of virtual machines," *Computer* , vol.38, no.5, pp. 32- 38, May 2005

Chapter 8.3 Textbook “Modern Operating Systems”

Abstraction & Virtualisation

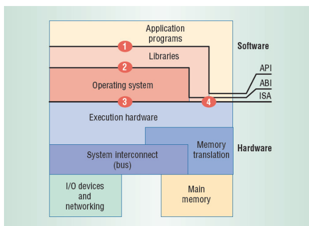


Interface Levels



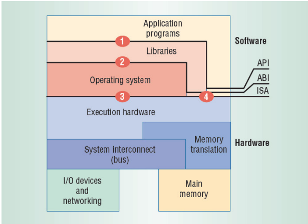
Instruction Set Architecture

- Interface between software and hardware
- Divided between privileged and un-privileged parts



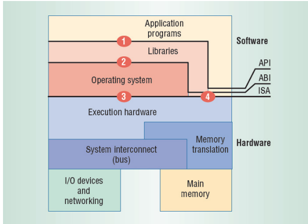
Application Binary Interface

- Interface between programs hardware + OS
- Consists of system call interface + un-privileged ISA

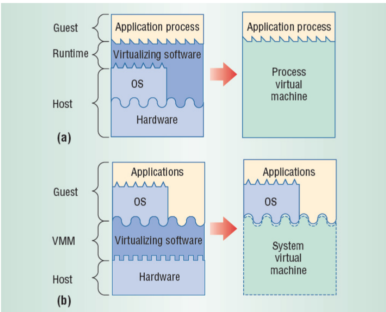


Application Programming Interface

- Interface between programs hardware + OS
- Consists of library calls + un-privileged ISA
  - Syscalls usually called through library.



Process versus *System* Virtual Machine



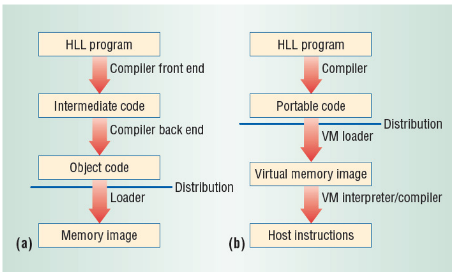
OS is an extended virtual machine

- Multiplexes the “machine” between applications
  - Time sharing, multitasking, batching
- Provided a higher-level machine for
  - Ease of use
  - Portability
  - Efficiency
  - Security
  - Etc.…

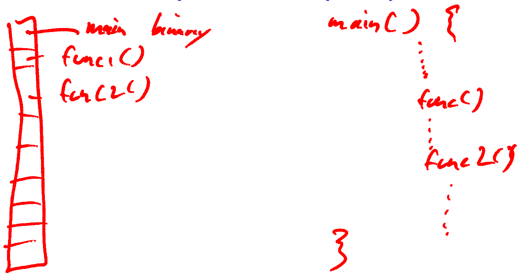
JAVA – Higher-level Virtual Machine

- write a program once, and run it anywhere
  - Architecture independent
  - Operating System independent
- Language itself was clean, robust, garbage collection
- Program compiled into bytecode
  - Interpreted or just-in-time compiled.
  - Lower than native performance

Conventional versus Emulation/Translation



Aside: Just In-Time compilation (JIT)



Issues

- Legacy applications
- No isolation nor resource management between applets
- Security
  - Trust JVM implementation? Trust underlying OS?
- Performance compared to native

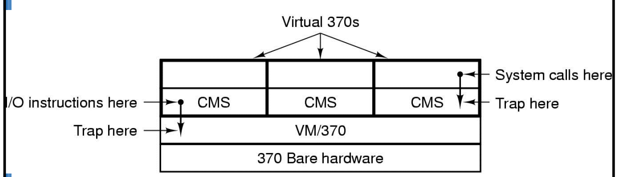
Is the OS the “right” level of extended machine?

- Security
  - Trust the underlying OS?
- Legacy application and OSs
- Resource management of existing systems suitable for all applications?
- What about activities requiring “root” privileges

Virtual Machine Monitors

- Provide scheduling and resource management
- Extended “machine” is the actual machine interface.

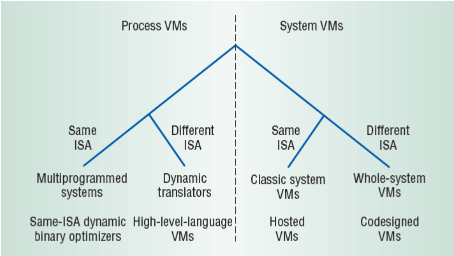
IBM VM/370



Advantages

- Legacy OSes (and applications)
- Server consolidation
- Concurrent OSes
  - Linux – Windows
  - Primary – Backup
    - High availability
- Test and Development
- Security
  - VMM (hopefully) small and correct
- Performance near bare hardware
  - For some applications

cse



THE UNIVERSITY OF NEW SOUTH WALES

cse

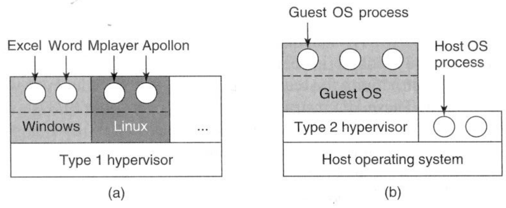


Figure 1-29. (a) A type 1 hypervisor. (b) A type 2 hypervisor.

THE UNIVERSITY OF NEW SOUTH WALES

cse

Virtual R3000???

- Interpret
  - System/161
    - slow
  - JIT dynamic compilation
- Run on the real hardware??

THE UNIVERSITY OF NEW SOUTH WALES

cse

Gerald J. Popek and Robert P. Goldberg (1974). "Formal Requirements for Virtualizable Third Generation Architectures". Communications of the ACM 17 (7): 412 –421.

- Sensitive Instructions
  - The instructions that attempt to change the configuration of the processor.
  - The instructions whose behaviour or result depends on the configuration of the processor.
- Privileged Instructions
  - Instructions that trap if the processor is in user mode and do not trap if it is in system mode.
- Theorem
  - Architecture is virtualisable if sensitive instructions are a subset of privileged instructions.

THE UNIVERSITY OF NEW SOUTH WALES

cse

R3000 Virtual Memory Addressing

- MMU
  - address translation in hardware
  - management of translation is software

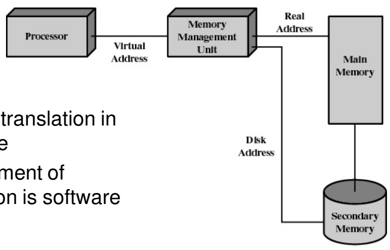


Figure 2.10 Virtual Memory Addressing

23

THE UNIVERSITY OF NEW SOUTH WALES

cse

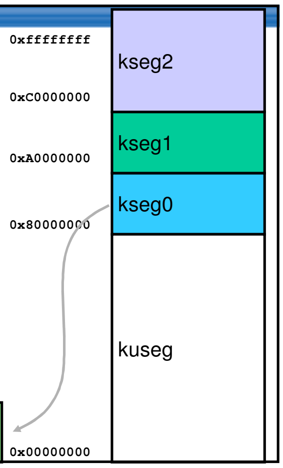
R3000 Address Space Layout

- kuseg:
  - 2 gigabytes
  - MMU translated
  - Cacheable
  - user-mode and kernel mode accessible

THE UNIVERSITY OF NEW SOUTH WALES

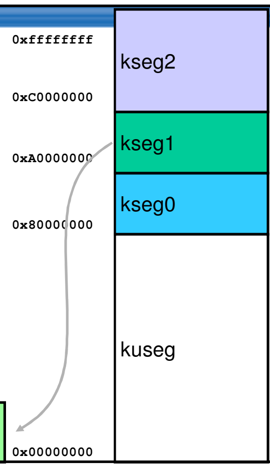
R3000 Address Space Layout

- kseg0:
  - 512 megabytes
  - Fixed translation window to physical memory
    - 0x80000000 - 0x9fffffff virtual = 0x00000000 - 0x1fffffff physical
    - MMU not used
  - Cacheable
  - Only kernel-mode accessible
  - Usually where the kernel code is placed



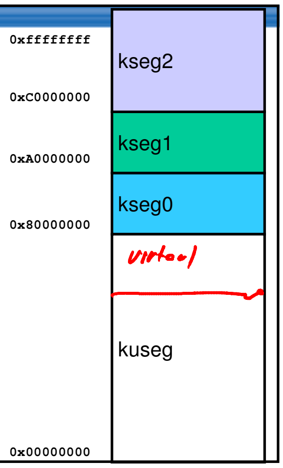
R3000 Address Space Layout

- kseg1:
  - 512 megabytes
  - Fixed translation window to physical memory
    - 0xa0000000 - 0xbfffffff virtual = 0x00000000 - 0x1fffffff physical
    - MMU not used
  - **NOT** cacheable
  - Only kernel-mode accessible
  - Where devices are accessed (and boot ROM)



R3000 Address Space Layout

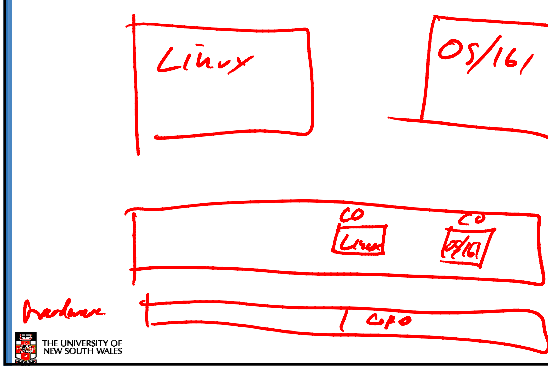
- kseg2:
  - 1024 megabytes
  - MMU translated
  - Cacheable
  - Only kernel-mode accessible



Issues

- Privileged registers (CP0)
- Privileged instructions
- Address Spaces
- Exceptions (including syscalls, interrupts)
- Devices

Approach: Trap & Emulate?



cse

THE UNIVERSITY OF NEW SOUTH WALES

cse

THE UNIVERSITY OF NEW SOUTH WALES

cse

THE UNIVERSITY OF NEW SOUTH WALES