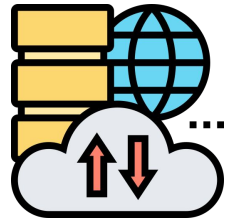
COEN 241
Introduction to Cloud Computing

Lecture 3 - Advanced Virtualization Concepts I

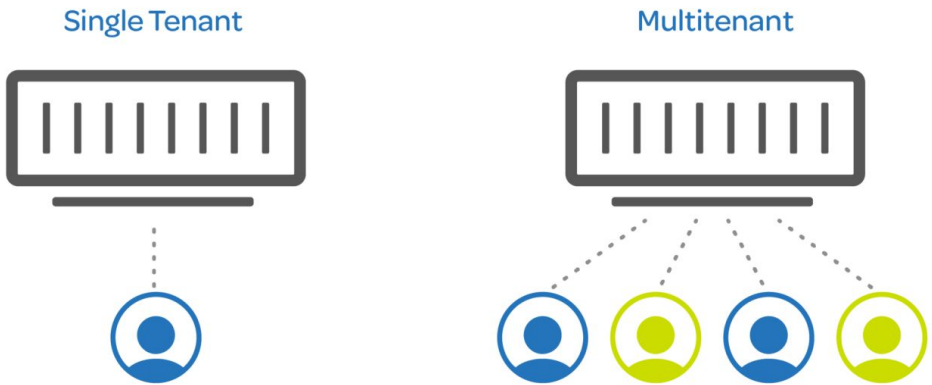


Lecture 2 Recap

- Multitenancy
- Virtualizations

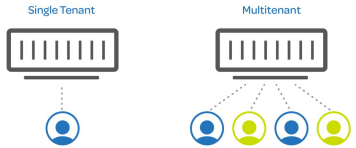
Multitenancy

- Definition: An architecture where multiple users share a single (software or hardware) resource



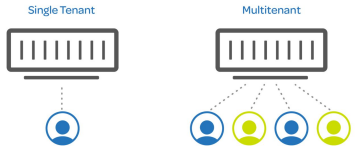
Multitenancy

- Benefits
  - Cost efficient
  - Easier to monitor and manage
  - Easily scalable
- Drawbacks
  - Generally more complex architecture
  - Security and isolation
  - Allocation



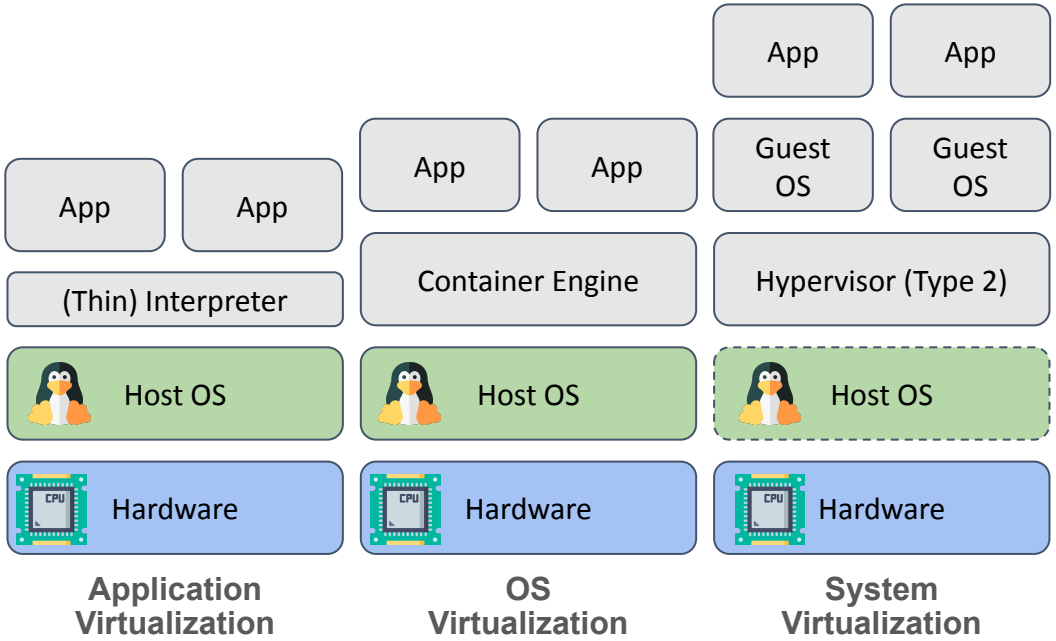
How to Enable Multitenancy?

- Virtualization!
- Definition: Using **software** to create an abstraction layer over the **hardware** that allows the hardware elements of a single computer—processors, memory, storage and more—**to be divided into multiple virtual elements**.



Types of Virtualization

- **Application Virtualization**
  - E.g., JVM, PVM
- **OS Virtualization**
  - Containers
- **System Virtualization**
  - Virtual Machines
- Network Virtualization
- Storage Virtualization
- Many more...



Agenda for Today

- Project Idea Overview
- Advanced Virtualization Concepts
  - Hypervisor
    - Xen
  - CPU Support for Virtualization
- Simulation vs. Emulation vs. Virtualization
- QEMU & Demo
- Readings
  - Recommended: None
  - Optional
    - https://wiki.xenproject.org/wiki/Xen_Project_Software_Overview
    - https://www.cl.cam.ac.uk/research/srg/netos/papers/2003-xensosp.pdf

Project Ideas

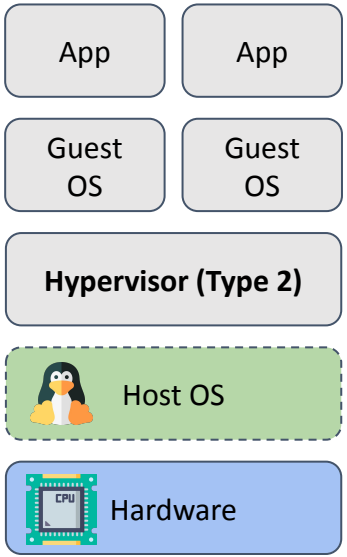
Project Ideas from Previous Classes

- Serverless Restaurant Reservation
- Recommendation System
- Cloud based emulation
- Useful apps for school life

Hypervisor

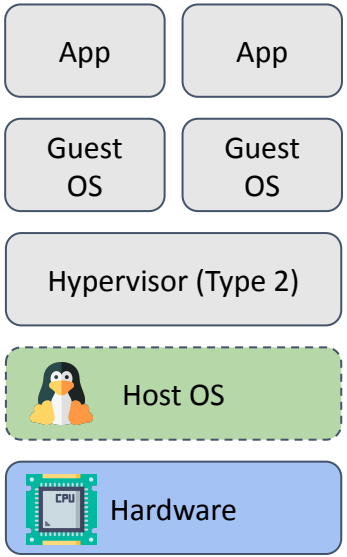
Hypervisor (Virtual Machine Monitor) Recap

- A thin layer of software that's between the hardware and the operating system, virtualizing and managing all hardware resources for VMs
- Examples
  - VMWare
  - Virtualbox
  - Parallels
  - QEMU
  - **Xen**
  - KVM
- Generally does not provide OS functionalities
- Just enough features to isolate the VMs



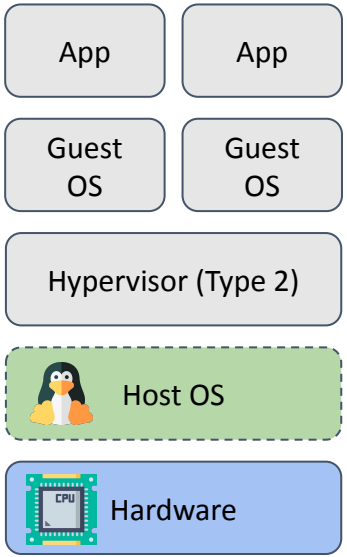
Type 1 & Type 2 Hypervisor

- Type 1 Hypervisor
  - Runs directly on the host's hardware
  - Used for most production environment
  - E.g., Citrix/Xen Server, VMware ESXi, Microsoft Hyper-V
  - Pros: Secure, performant
  - Cons: Require hardware support, harder to use
- Type 2 Hypervisor
  - Runs on an operating system similar to other processes
  - E.g., Oracle Virtual Box, VMware Workstation
  - Pro: Easier to use (We will use this for HW1)
  - Con: Less performant and less secure



Hypervisor Capabilities

- Manages CPU, RAM and device I/O (input/output)
  - Device I/O include disk, network, graphics, USB, ...
- Share concepts with microkernel operating systems
  - i.e., microkernel can't do the complete job of an operating system
- For device I/O, must prevent guests “breaking out”
- Base security approach on per-device capabilities:
  - Network card might be VM-aware and can isolate functions
  - Need to be very careful about capabilities such as DMA
    - Direct Memory Access lets devices read/write memory without CPU



Recap: History of System Virtualization

- Early x86 computers were underpowered
  - Windows 95 with few applications often maxed-out resources
  - Server work left to expensive server-class computers
- PC computing power increased
  - X86 servers began to appear
  - Mostly under-utilized due to no virtualization
- **x86 virtualization surged mid-2000s thanks to the Xen hypervisor**

Digression: Xen Project

- "XenoServers" research at University of Cambridge
  - Goal: allow computing resources to migrate
  - E.g.: Running a gaming server to migrate near the players to reduce latency
- But the project needed a means to migrate servers
  - Existing solutions like VMWare were proprietary and costly
- Needed to build a new Hypervisor!

Digression: Xen Project

- Xen was presented in 2003 SOSP
  - One of the top CS Conferences
  - https://www.cl.cam.ac.uk/research/srg/netos/papers/2003-xensosp.pdf
- Xen (then) required para-virtualised OSs as VMs
  - We will discuss paravirtualization soon
- Demonstrated on both Linux (XenoLinux) and Windows XP
  - Was much faster than VMWare
  - Able to scale to 100 VMs in a single commodity machine
  - Open Source

Digression: Xen Project

- Xen was super fast! 92% of native Linux speed

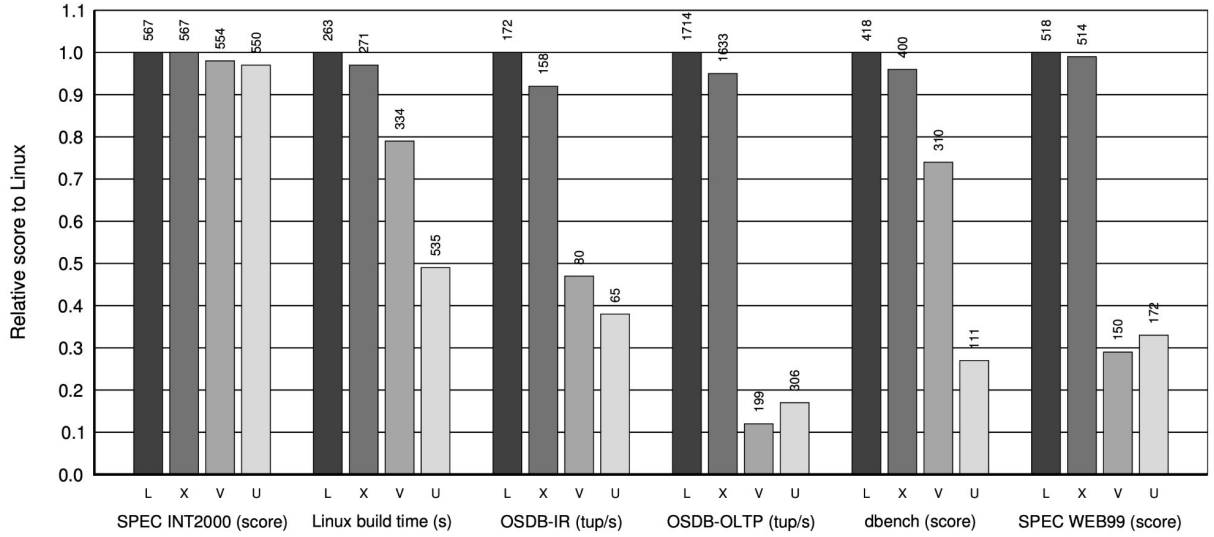
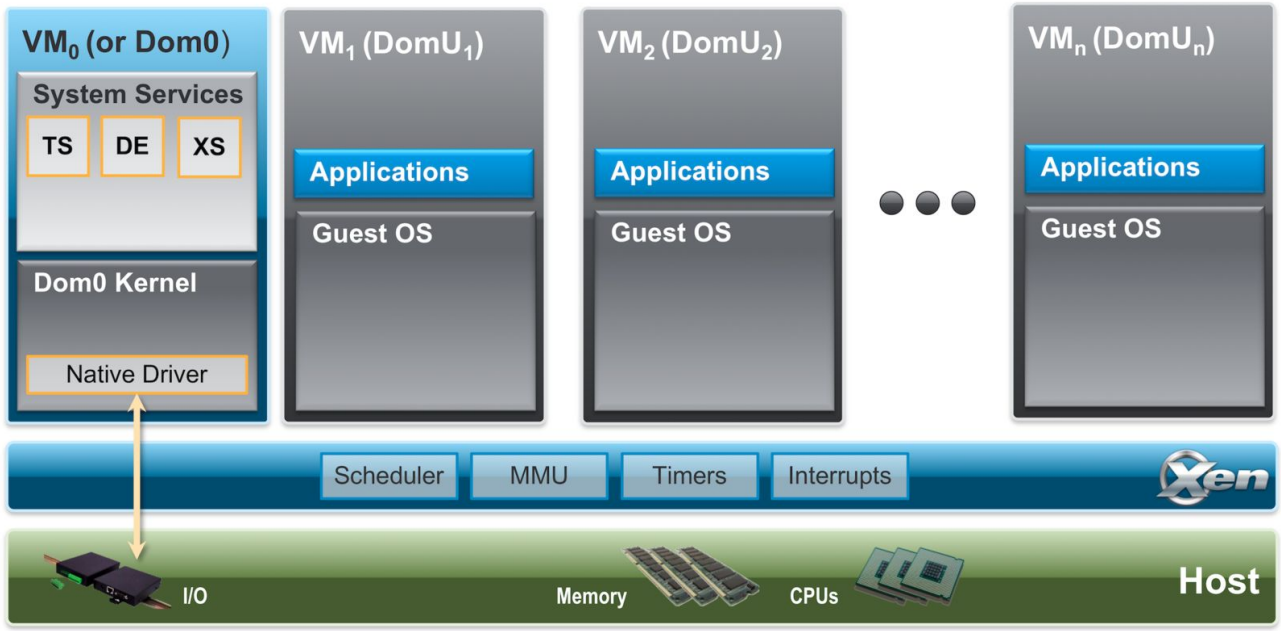


Figure 3: Relative performance of native Linux (L), XenoLinux (X), VMware workstation 3.2 (V) and User-Mode Linux (U).

Digression: Xen & Amazon

- Amazon was growing a new EC2 business
  - Needed a fast and optimal hypervisor for their servers
- Xen was a perfect choice for Amazon
  - Cost efficient (Open Source) vs VMWare or IBM
  - Good Performance
  - Heavily used until 2017
- Xen founders were not left out of pocket
  - Founded XenSource company to provide Xen support
  - XenSource was bought out by Citrix

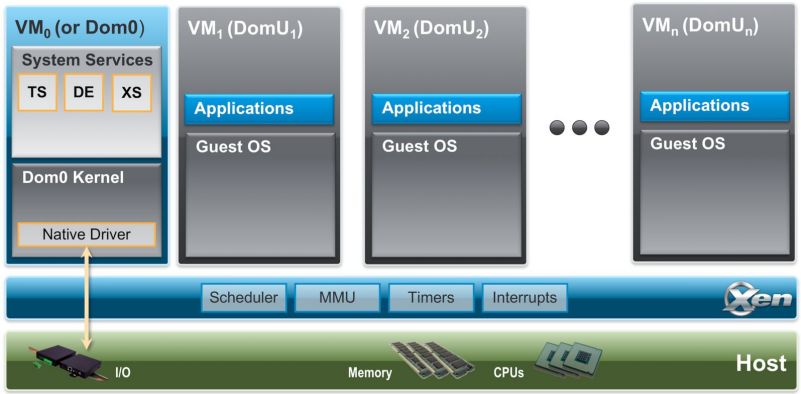
Xen Architecture



https://wiki.xenproject.org/wiki/File:Xen_Arch_Diagram_v2.png

Xen Architecture

- Xen divides hosts into Domains
  - Dom0 is special: the Linux layer that controls actual hardware
  - Then the DomUs can make ***hypercalls*** to access hardware to Dom0
  - Xen hypervisor delegates hardware interaction to Dom0
  - How is this different from regular hypervisors?



Hypervisor Security

- Users expect complete isolation
  - As if each VM is on a separate computer
- Aggregation must result in resource sharing
  - Security risk in control interactions with the hypervisor
  - Hypervisor management commands have exposed security holes
- Hypervisor must be designed to have a **small attack surface area**
  - Kernels have a large attack area, and are hard to secure
  - **Hyperjacking**: taking over (hijacking) the hypervisor from within a guest
- A Hard Problem!

Prior Hypervisor Security Holes

- VENOM (Virtualized Environment Neglected Operations Manipulation)
  - Problem with the floppy disk controller in QEMU
    - VirtualBox, Xen and KVM also used QEMU's code
  - Basically, guest accesses "floppy drive" via "I/O port"
  - QEMU driver keeps track of floppy drive commands in a buffer
  - but specially crafted requests could overflow buffer
  - Malicious VM can then take control of QEMU system
- Spectre & Meltdown
  - CVE-2017-5715, 2017-5753, 2017-5754
  - https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-5715

Live Migration

- Live migration—move running VM to another host
  - A good demonstration that your hypervisor is efficient!
  - “Live” usually means no detectable downtime
  - i.e., cannot pause VM, copy VM state, resume VM on new host
  - Repeatedly stream memory updates until can do switch-over
- Requirements of physical hosts supporting migration:
  - NICs are receiving the same MAC address
  - Simplest for storage to be network-based (e.g., iSCSI / NFS)

High Availability

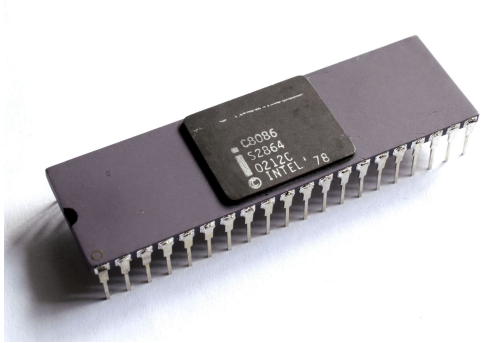
- High availability (HA) means VMs are robust to failure:
  - VMs may restart on the same host (e.g., given typical OS crash)
  - Or host may fail: ensure VMs can continue on a different host
- Live migration & high availability have common needs
  - HA requires VM that might take over being up-to-date
  - Similar to a persistent live migration from leader to follower
- Care needed to ensure safe and consistent failover

CPU Hardware Support for Virtualization

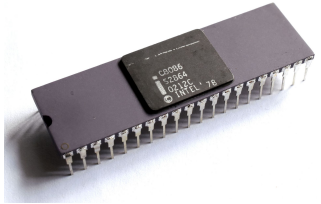
Commodity Hardware for Cloud

- Cloud is generally built with cheap commodity hardware
  - Different from running expensive large computers in the past
- Distributed coordination in software is more important than server hardware reliability, so we can use cheap servers
- Traditional Commodity x86 CPUs **were not built for virtualization**
  - Pretty slow, as everything was software based

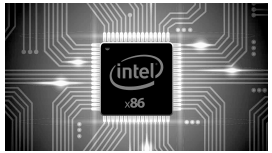
History of x86 Chips



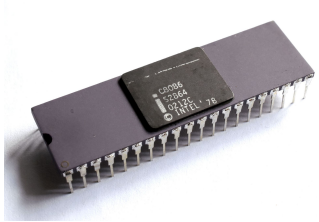
History of x86 Chips



- First x86, Intel 8086 (1978) the IBM PC CPU, had no protected mode
  - Failures in a single application could take down the whole OS!
- Intel 80**286** (1982) booted real mode; added protected mode
  - Real address mode & protected virtual-address mode
  - Could not revert from protected mode to the basic 8086-compatible real address mode without hardware reset
- Intel 80**386** (1985) introduced virtual 8086 mode
  - Allowed running virtual 8086 environments, e.g., MS-DOS on Win 3.1

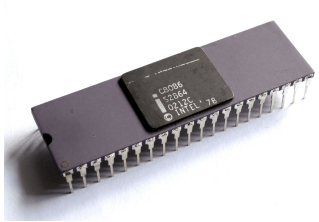


CPU Modes



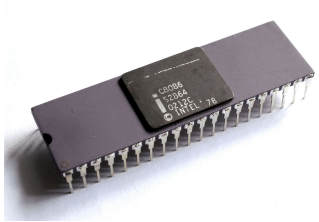
- Operating modes place restrictions on the type and scope of operations that can be performed by certain processes being run by the CPU
  - Also called processor modes, CPU states, CPU privilege levels and other name
  - Allows the OS to run with more privileges than application software
- Ways that the processor creates an operating environment for itself.
- Specifically, the processor mode controls how the processor sees and manages the system memory and the tasks that use it

CPU Modes

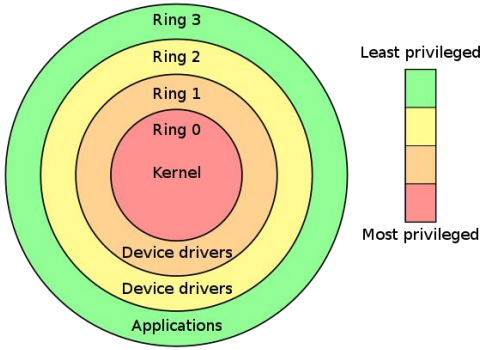


- The 80386 has three processing modes (grouped as legacy mode):
  - Protected Mode.
  - Real-Address Mode.
  - Virtual 8086 Mode.
- New 64-bit CPU (AMD64) has what is called long-mode
- Example Differences between Long and Protected Mode
  - Protected Mode: The page table entry was only four bytes long, so you had 1024 entries per table (for a 4KB page table)
  - Long Mode: The page table entry is 8 bytes long, so you had 512 entries per table

Real Mode

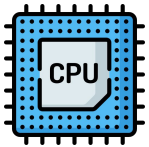


- Real(-address) mode is the mode of the processor immediately after RESET
- 80386 appears to programmers as a fast 8086 with some new instructions.
- Now mainly used for bootloader only
  - Performs memory map detection, detection of available video modes, loading of additional files, etc
  - What is a bootloader and why use real mode here?



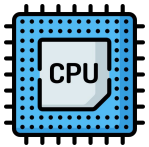
Protected Mode

- Natural 32-bit environment of the 80386 processor
- Permits system software / OS to use features such as virtual memory, paging and safe multi-tasking
- Enables memory protection, privilege levels called rings
  - Allows OSs to set up CPU to isolate kernel and userspace
- Did not play well with software based virtualization
  - Fancy techniques like ring deprivileging needed to be used
    - Guest OS in higher ring than host OS



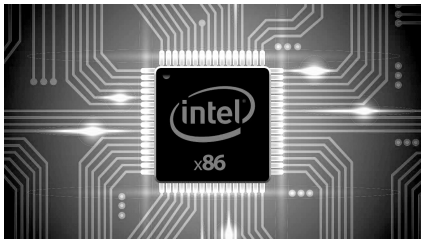
Virtual 8086 Mode

- A dynamic mode where the processor can switch repeatedly and rapidly between V86 mode and protected mode
- The CPU enters V86 mode from protected mode to execute an 8086 program, then leaves V86 mode and enters protected mode to continue executing a native 80386 program.
- https://en.wikipedia.org/wiki/X86-64#Physical_address_space_details



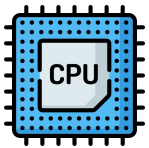
Challenges in x86/x64 Virtualization

- CPU **protected mode** and CPU **long mode** (64-bit operation)
  - These modes weren't designed without virtualization in mind
- 20 years between first CPU and the first hardware virtualization efforts
  - Virtual Mode arrived after 20 years
- Hidden CPU state
  - VMM can't save/restore this state later when switching VMs
- Memory management inefficiency
- I/O interactions
  - Again, designed without virtualization in mind



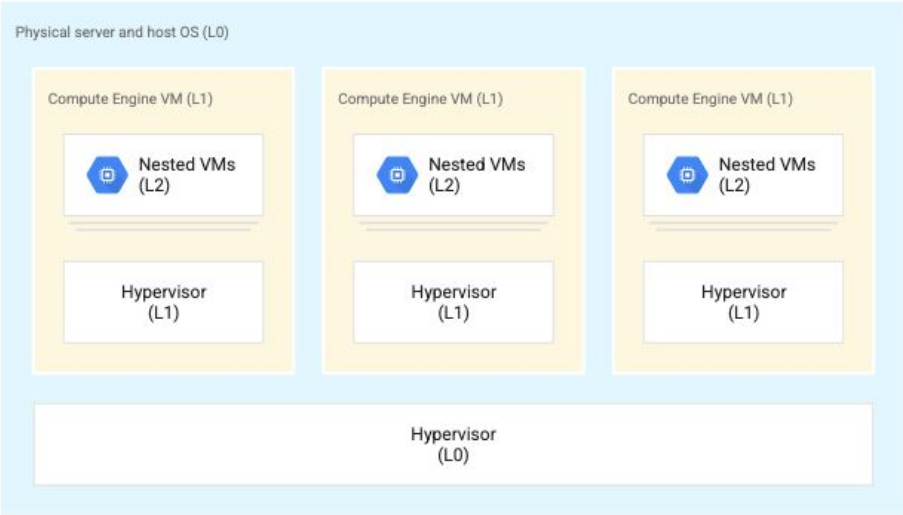
Hardware Virtualization Support in x86/x64

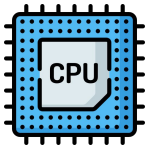
- CPUs gain a guest mode within protected mode
  - For guest OSes, guest mode looks like protected mode
  - For hosts, guest mode is lower privilege than protected mode
- Intel VT-x released in 2005 for some Pentium 4 CPUs
  - Virtual Machine Extention and adds new instructions for virtualization
  - Subsequent CPUs include it (except some Atom processors)
  - AMD released an equivalent technology in 2006 (AMD-v)
- More memory virtualization support still to come…



Nested Virtualization

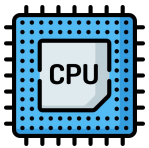
- Running VMs inside another VM





Nested Virtualization: Use Cases

- You have special VMs that you can't run on a provider’s infrastructure
- You have a software you use to test and validate new versions of a software package on numerous versions of different OSes



Digression: Is ARM the future?

- X86 vs ARM
  - Complex Instruction Set Computer vs Reduced Instruction Set Computer
  - More Hardware Control vs More Software Control
  - Traditionally for Personal Computers and Servers vs Mobile
- ARM is starting to appear in PCs and modern data centers!
  - Apple M1
  - AWS
- Virtualization on ARM is different

Simulation vs. Emulation vs. Virtualization

Simulation vs. Emulation vs. Virtualization

- **Simulation**
  - Running a model of some system to observe its behavior
  - Often not real-time
- **Emulation**
  - Originally described hardware-assisted simulation
  - Now used to mean a machine imitating another machine **using software**
  - Must be interactive
  - Often confused with virtualization
- **Virtualization**
  - Adding a supervisory layer to an existing system
  - Can **access the hardware directly**

Emulation vs. Virtualization

Emulation	Virtualization
Software emulator needed to access the hardware	Can access the hardware directly
Need an interpreter to run some code	Can run the code directly
Relatively Slower	Relatively Faster
Relatively costlier	Relatively more expensive
Relatively harder to backup and recover	Better backup and recovery

QEMU

QEMU

- Open source **emulator and virtualization solution**
  - CPU hosting is emulation rather than virtualization
  - QEMU aims to run as much of the guest system's code on the actual host CPU as possible
- Supports multiple CPU types due to emulation
  - For non-native CPUs, dynamic binary translation cross-compiles guest machine code into code the host CPU can run

QEMU

- QEMU's software components used in VirtualBox
- QEMU defines formats of disk images—e.g., qcow2
  - These are files that represent, e.g., virtual hard disks
- QEMU implemented many devices / subsystems:
  - PIIX3 IDE for interacting with virtual devices like hard-disks
  - VGA emulator
  - Common network interface card emulation, e.g., R1000
  - ACPI support

QEMU Demo!

QEMU Demo Details

- Download & Install QEMU
- Create QEMU Image
- Install a Ubuntu VM

Agenda for Today

- Project Ideas Overview
- Advanced Virtualization Concepts
  - Hypervisor
    - Xen
  - CPU Support for Virtualization
- Simulation vs. Emulation vs. Virtualization
- QEMU & Demo
- Readings
  - Recommended: None
  - Optional
    - https://wiki.xenproject.org/wiki/Xen_Project_Software_Overview
    - https://www.cl.cam.ac.uk/research/srg/netos/papers/2003-xensosp.pdf

TODOs!

- Project Teams!
- HW 1 will be released after class!
  - You can start looking into it, we will do a preview next class
- Quiz 1 will be released after next class

Questions?