

YOU EVER WONDER WHY WE'RE HERE?

My First Existential Crisis - The Talk!



THE 1960S WAS A CRAZY TIME...

- Bunch of missile and political stuff happened
- Smoking was found to be “hazardous to your health”
- Star Trek Debuts
- The question “How do I make a computer do two things at a time” was asked

HOW DO I MAKE MY COMPUTER DO TWO THINGS AT ONE TIME?



BATCH PROCESSING

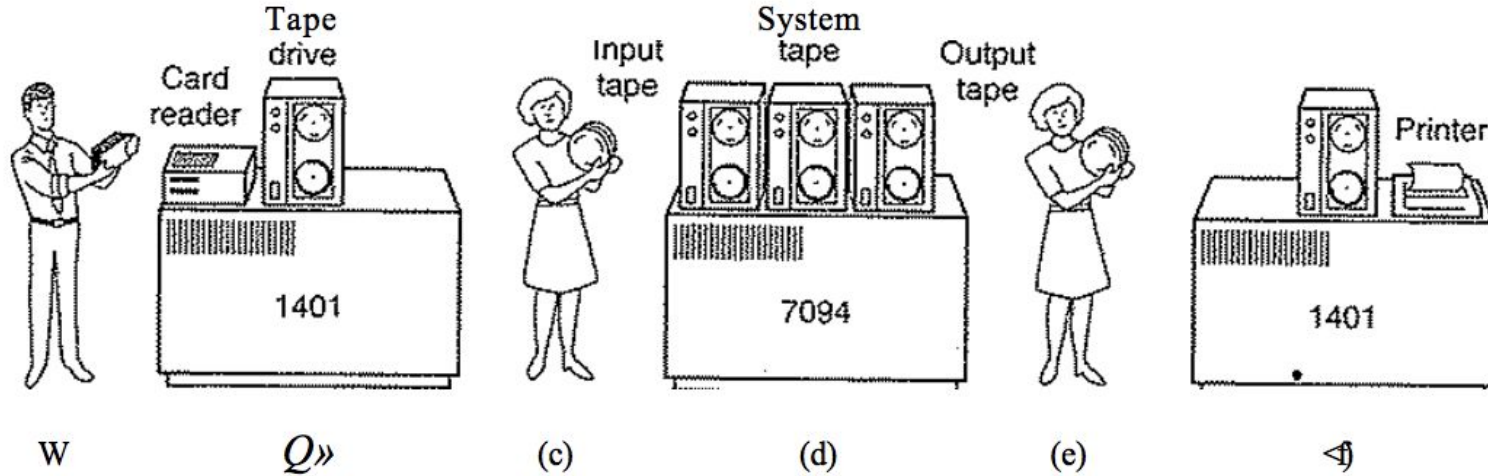
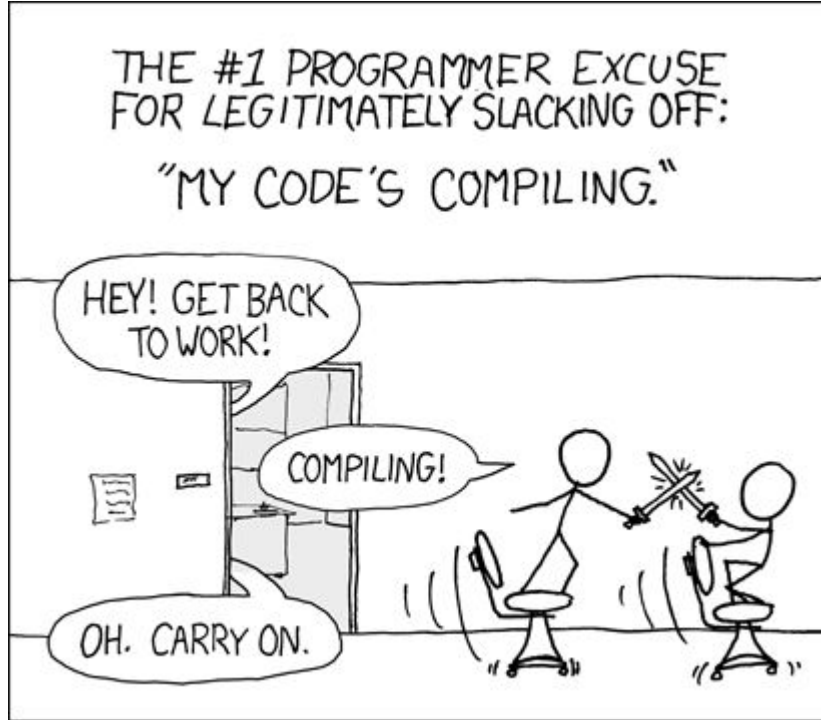


Figure 1-3. An early batch system, (a) Programmers bring cards to 1401. (b) 1401 reads batch of jobs onto tape, (c) Operator carries input tape to 7094. (d) 7094 does computing, (e) Operator carries output tape to 1401. (f) 1401 prints output.

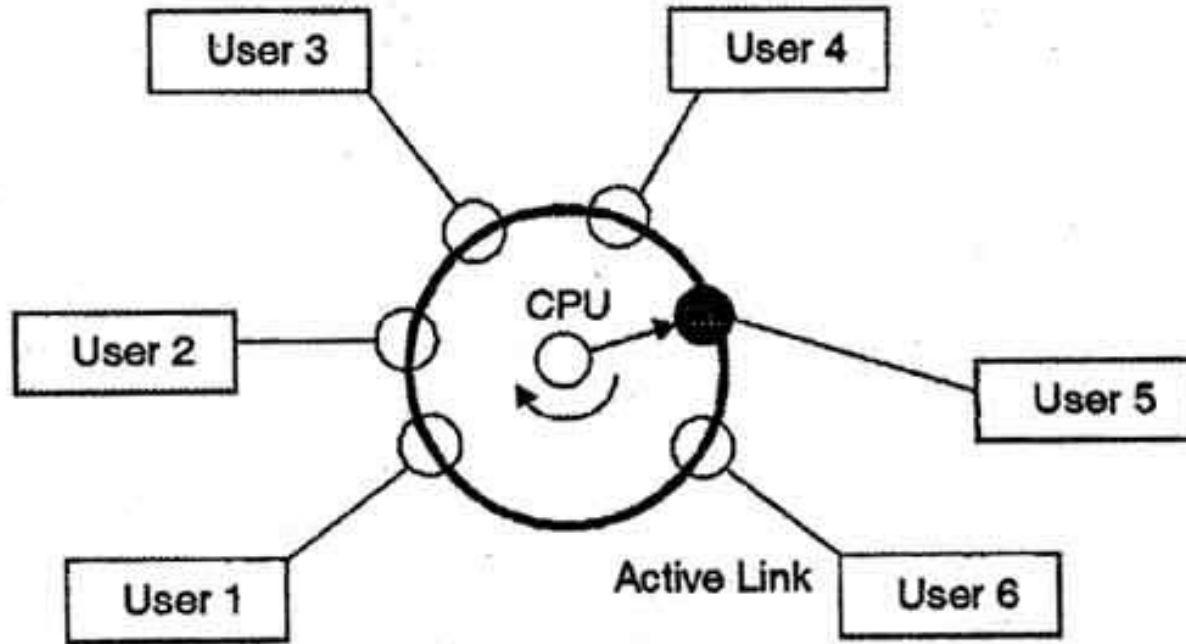
Tanenbaum, Andrew S. *Modern Operating Systems: Problem Solutions*. Prentice Hall, 1992.

BATCH PROCESSING

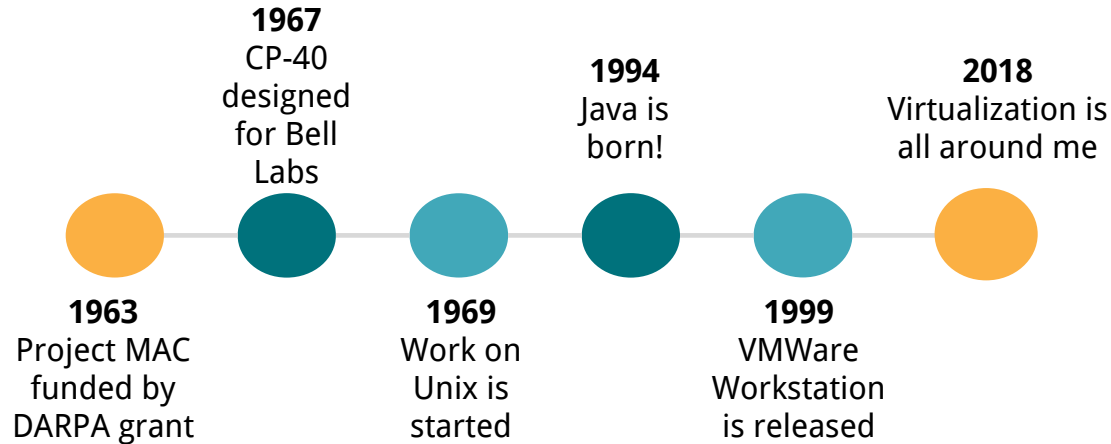


Source: <https://xkcd.com/303/>

TIME SHARING



WHERE DID IT BEGIN? - VIRTUAL MACHINES





"AN EFFICIENT, ISOLATED DUPLICATE OF
A REAL COMPUTER MACHINE" - POPEK AND GOLDBERG



LET'S BUILD OUR BASE REALLY QUICK

I know what a virtual machine is, but *someone else* might not

Emulation of your computer system

- Takes the underlying hardware and plays make believe by letting you configure it into what you want

A hypervisor takes care of the scheduling of these guest operating systems and only sees them as processes

- Used to be called Control Program or Virtual Machine Monitors (VMM)

"SEES THEM AS PROCESSES", THAT SOUNDS
FAMILIAR..



WORDS THAT I DIDN'T KNOW

So you may not too

User Mode

- Processor executes “normal” instructions
- When encountering a privileged instruction, processor switches to kernel mode to preform

Kernel Mode

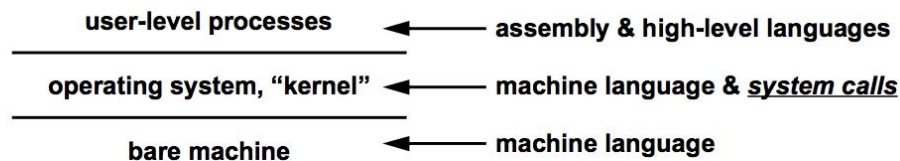
- Processor executes both normal and privileged instructions

WORDS THAT I DIDN'T KNOW

So you may not too

Privileged Instruction

- Something that needs to be ran in kernel mode
 - Initiate I/O
 - Switch state vectors or contexts
 - Load/save from protected memory



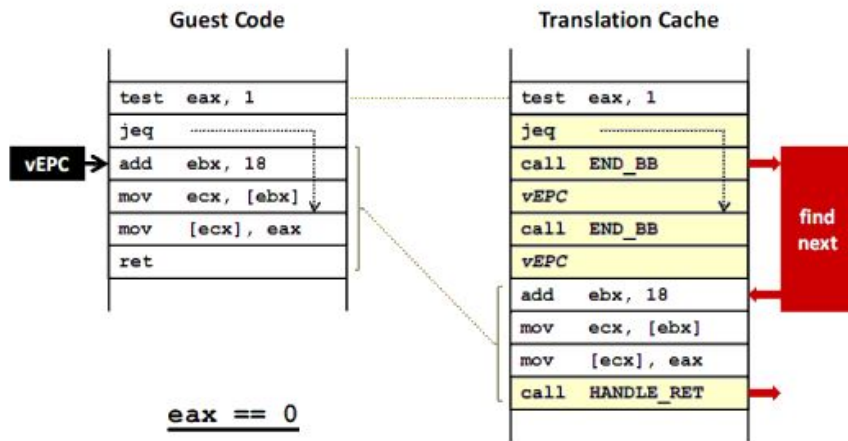
WORDS THAT I DIDN'T KNOW

So you may not too

Binary Translation

Takes the code that the Guest OS wants to run in Kernel Mode and translate it's to be safe for other guest OSes and the hypervisor

Controlling Control Flow



HOW DO THEY WORK?

Hold onto your seats..

Full Virtualization

- Guest OS thinks it is alone
- Host OS simulates what the Guest OS thinks is hardware
- Relies on binary translation to trap and execute sensitive instructions
- Example: VMWare

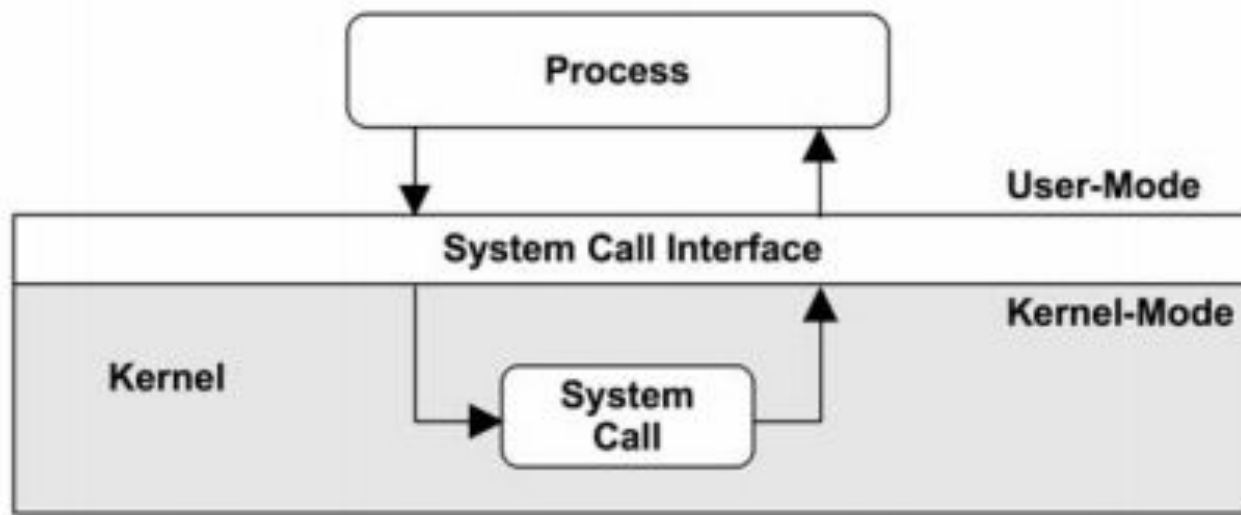
Paravirtualization

- Guest OS speaks to underlying hardware
- Enables virtualization on hardware that does not allow hardware assisted virtualization
- Example: Xen, xHyve

Hardware Assisted Virtualization

- Unmodified Guest
- Uses processor extensions to emulate privileged instructions in the guest
- Examples: VMWare or Xen on specific platforms

LET'S TAKE A LOOK



Source: <https://notes.shichao.io/spec/ch3/>

LET'S TAKE A LOOK

Type 1 Hypervisors

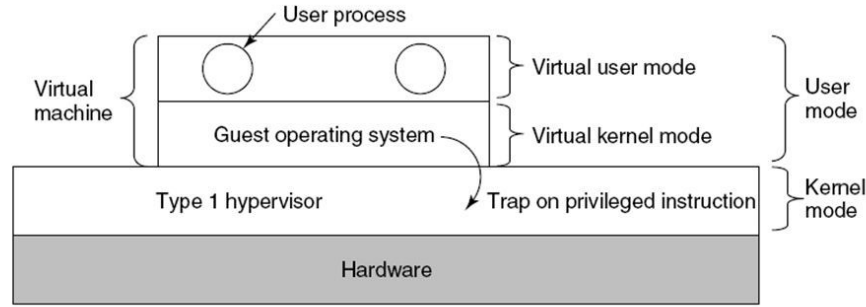
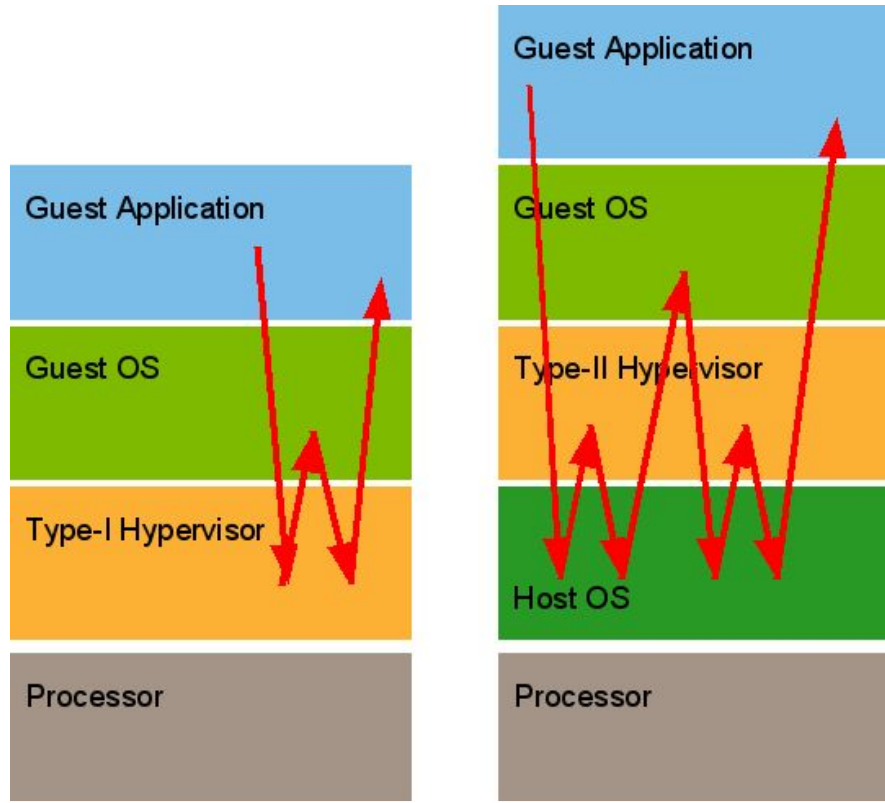


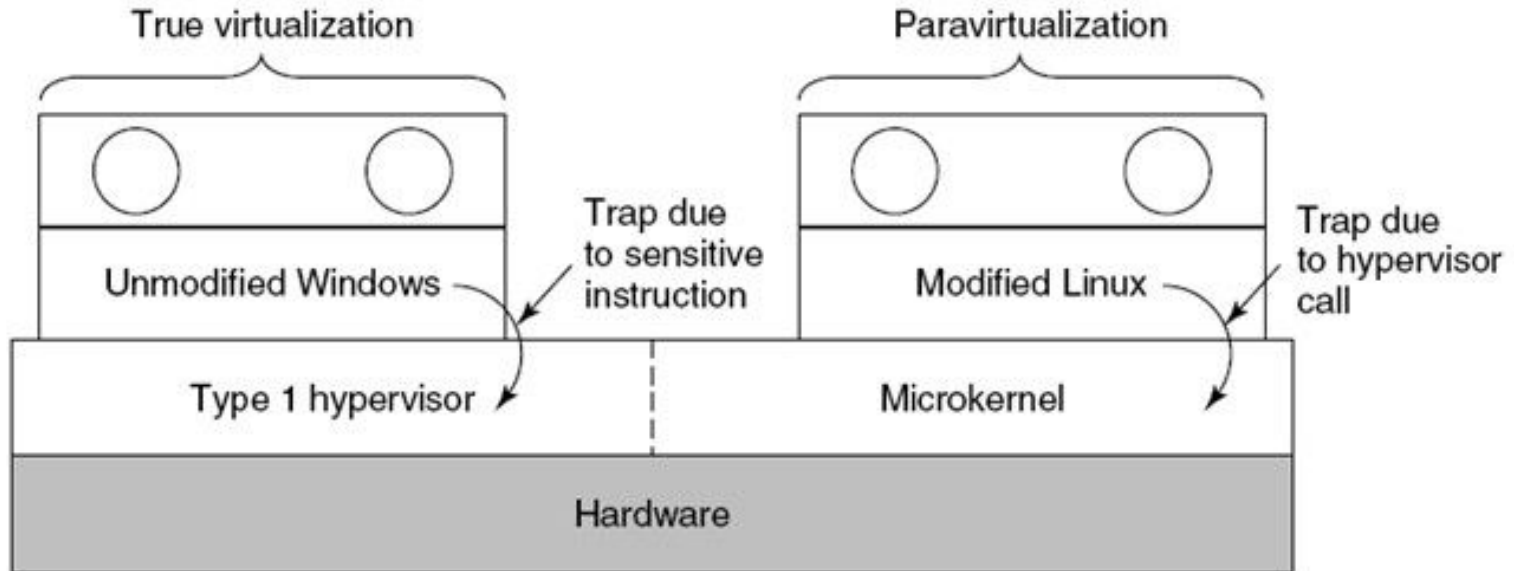
Figure 8-26. When the operating system in a virtual machine executes a kernel-only instruction, it traps to the hypervisor if virtualization technology is present.

LET'S TAKE A LOOK



Source: <https://microkerneldude.wordpress.com/2009/03/23/virtualization-some-get-it-some-dont/>

LET'S TAKE A LOOK



THERE ARE MANY TYPES OF VIRTUALIZATION

Full Virtualization, Paravirtualization, Software Virtualization, etc

VMS ARE AN EMULATION OF A MACHINE

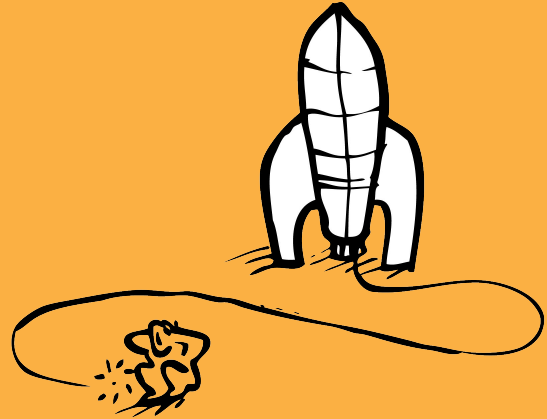
It takes the underlying hardware and allows it to be used in different ways

THERE'S A LOT GOING ON IN A VM

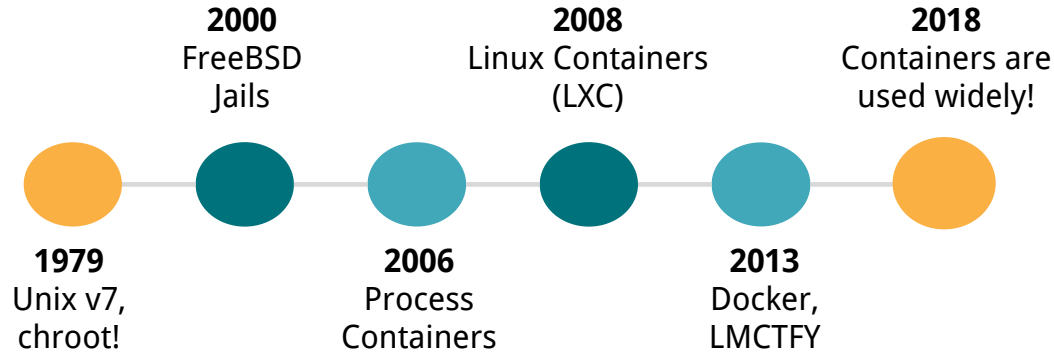
The amount of communication the hypervisor has to have is insane!

CONTAINERS ARE COOL

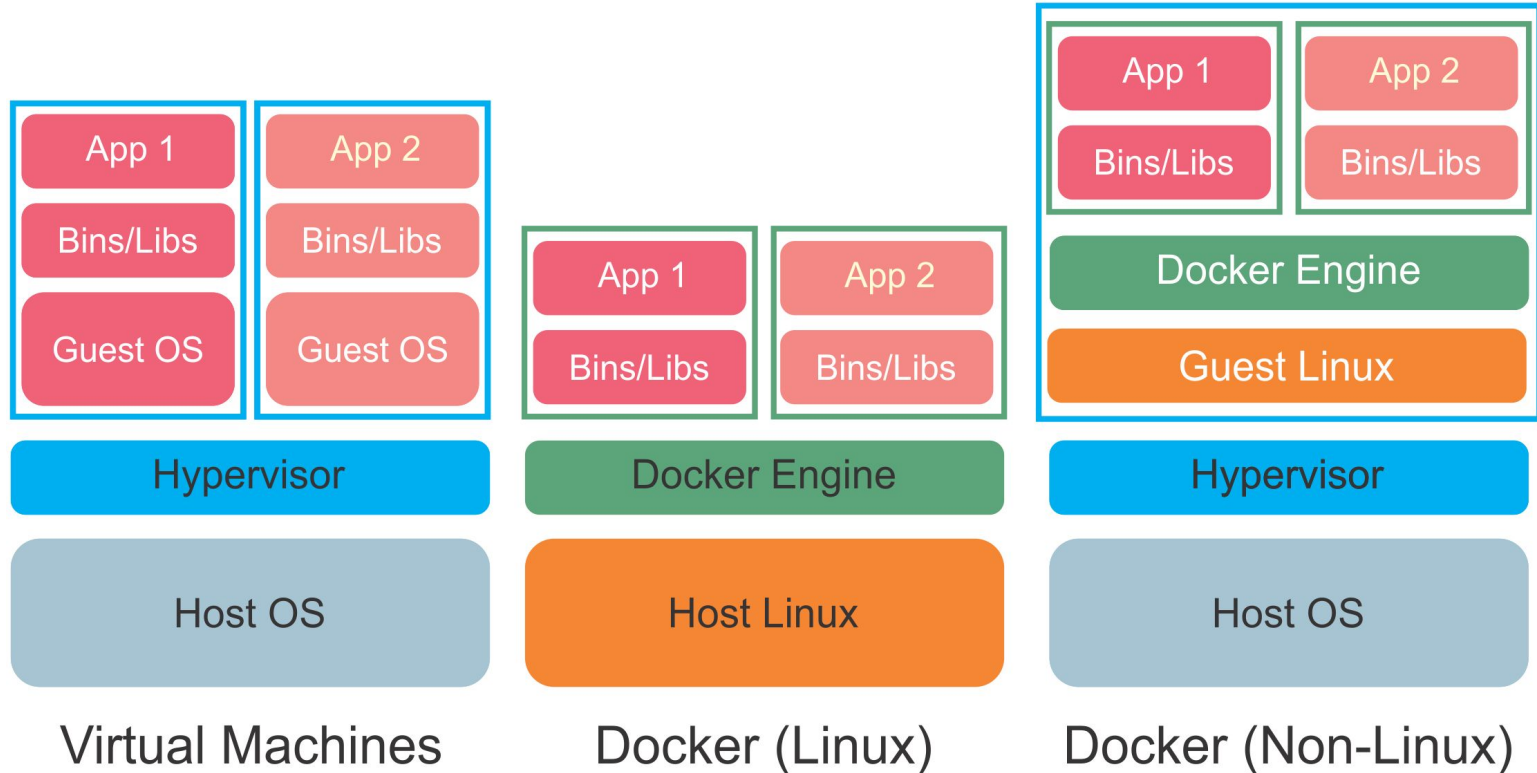
They're why we're here!



WHERE DID IT BEGIN? - CONTAINERS

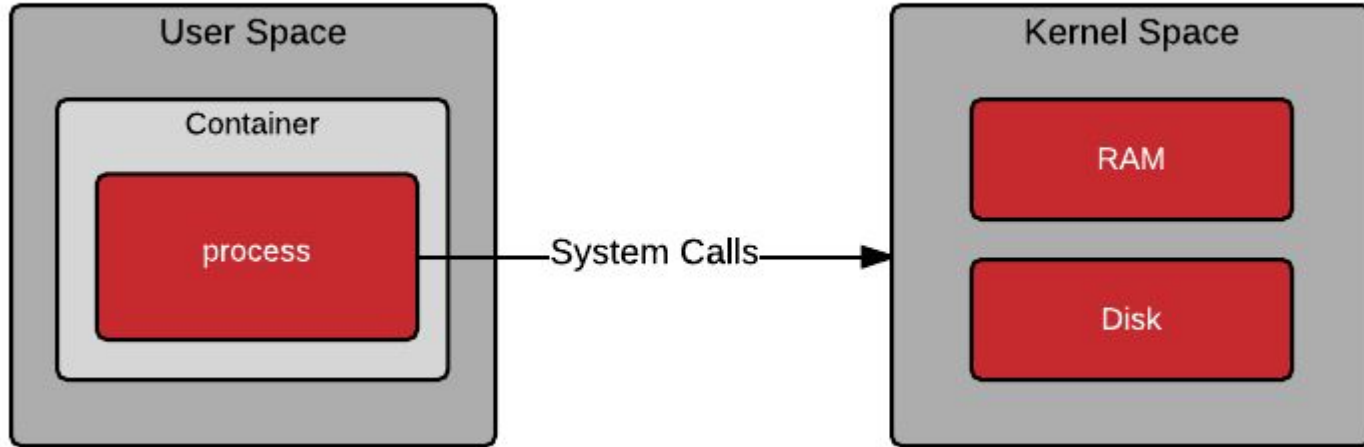


COMPARED TO WHAT WE'VE LEARNED...



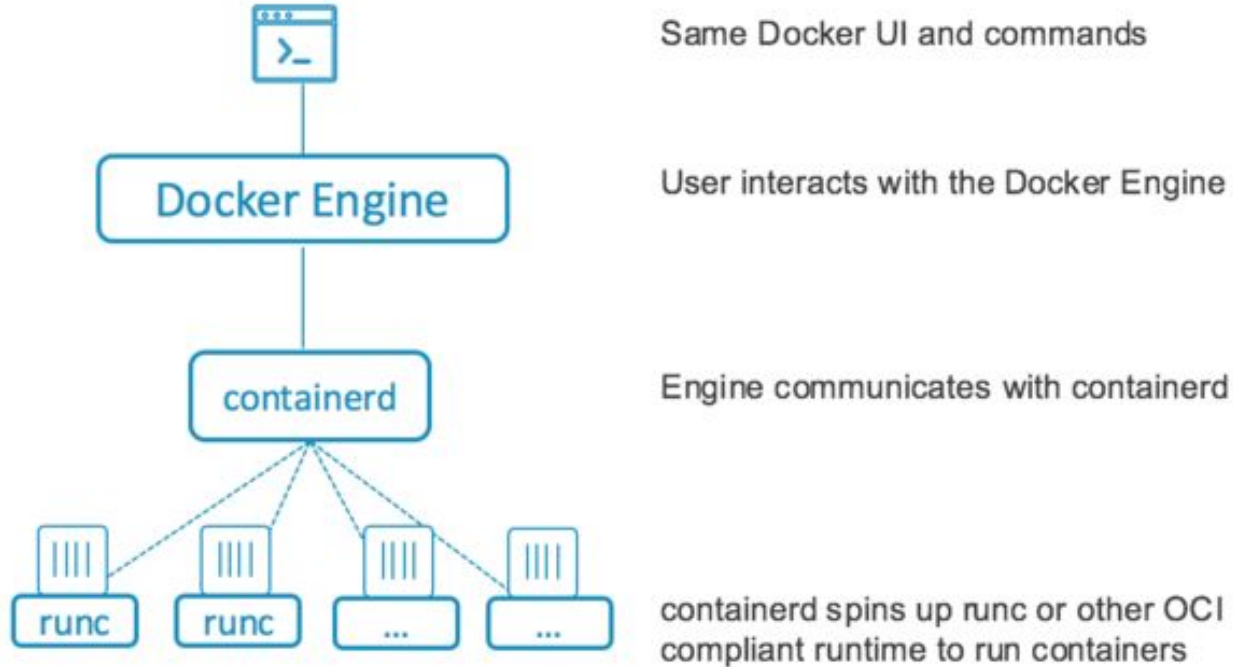
Source: <http://journals.plos.org/plosone/article?id=10.1371/journal.pone.0152686>

LET'S TAKE A LOOK



Source: <https://developers.redhat.com/blog/2018/02/22/container-terminology-practical-introduction/>

WHAT'S UNDER THE HOOD?



Source: <https://stackoverflow.com/questions/41645665/how-containerd-compares-to-runc>

WHAT DOES THIS STUFF DO

And why are there so many acronyms

Container Engine

- Easy to develop on
- API driven
- Abstracts the messy stuff to make containers fun!
- Docker, rkt

Container Runtime

- Manages container lifecycle
- containerd, rktlet
- CRI specific:
 - CRI-O
 - Dockershim
 - cri-containerd

Actual Containers

- Sometimes they don't close right
- Am I talking about tupperware or shipping containers?
- Good for holding leftovers that you'll never eat



WE'RE ALL LINUX IN THE END

Namespaces

Limits and virtualize the system resources as a collection of processes

- Mount
- Process ID (PID)
- Network
- IPC (InterProcess Communication)
- UTS
- User ID (UID)

Control Groups (cgroups)

Limits and isolates the resource usage as a collection of processes

- CPU
- Memory
- Disk I/O
- Network



CONTAINERS ARE NOT LIGHTWEIGHT VIRTUAL MACHINES



"CONTAINERS ARE AN EASY TO USE WRAPPERS
AROUND CGROUPS AND NAMESPACES" - JESS FRAZELLE (I THINK?)





VIRTUALIZE SYSTEMS
CONTAIN APPLICATIONS



CONTAINERS RELY ON CGROUPS AND NAMESPACES

Without them they really couldn't contain much

CONTAINERS ARE NOT LIGHTWEIGHT VMS

VMs will help you emulate, containers will help you ship and deploy!

CONTAINERS MAY NOT BE RIGHT FOR YOU!

Some applications cannot be cut into microservices, or your application may not benefit from containers.

KUBERNETES!

It's important too



WHAT IS IT?



Everyone needs a babysitter...

-



WHAT ARE THEIR STRENGTHS?

Swarm

- Made by Docker
- Tightly integrated with Docker
- Up to 1,000 nodes

Kubernetes

- Cluster management, scheduling, service discovery, and more!
- 1,000+ contributors
- Up to 5,000 nodes

Nomad

- Only provides cluster management and scheduling
- Follows Unix philosophy
- Simple - single binary
- 5,000+ nodes

CONTAIN APPLICATIONS, VIRTUALIZE SYSTEMS

CONTAINERS != LIGHTWEIGHT VMS

CONTAINERS MAY NOT BE RIGHT FOR YOU

WHEN DO I USE THIS?

Examples please!





QUESTIONS? COMMENTS? FEEDBACK?

tinyurl.com/kubeconqa