Virtualization and Abstractions

CS 240 - The University of Illinois

Wade Fagen-Ulmschneider

October 12, 2021

Operating Systems: A Great Illusionist

★ **So far, the Operating System has done an amazing job:**

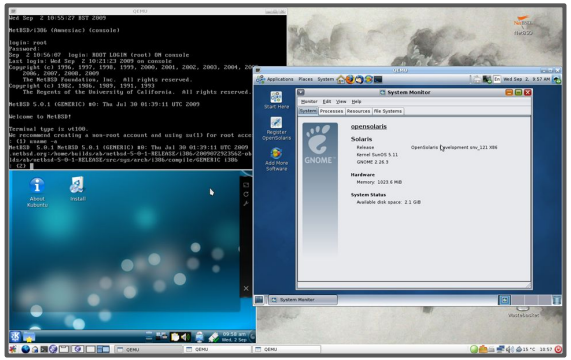
- As a process, it appears that we have _____.
- ...and has _____!

★ Do we really need more abstraction??

Big Idea: The OS is an illusionist

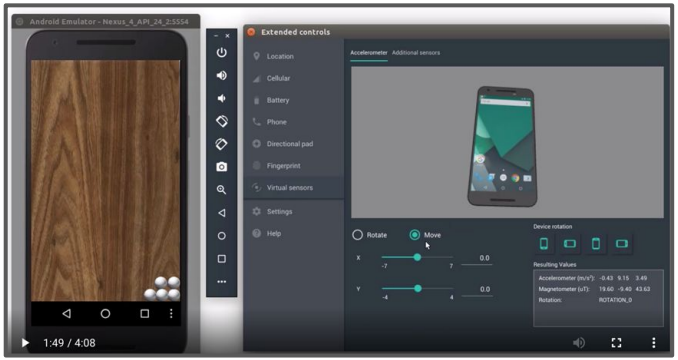
Hardware Platform Virtualization

Running hardware platform-specific binaries on different hardware.



Operating System Virtualization

Running guest operating systems within a host operating system environment (VirtualBox)



Hardware Virtualization

Mobile development is full of hardware virtualization to test mobile apps in various environments.

Virtualization

★ The goal of all virtualization is to map a _____ onto a _____:

Virtualization

★ The goal of all virtualization is to map a **virtual machine** onto a **host machine**:

- All virtual states $\mathbf{S_x}$ can be represented on the host system as $\mathbf{H(S_x)}$.
- For all sequence of translations between $\mathbf{S_1 \Rightarrow S_2}$, there's a sequence of operations that map $\mathbf{H(S_1) \Rightarrow H(S_2)}$.

A Virtual “Machine”

★ A “**machine**” is:

A Virtual “Machine”

★ A “**machine**” is: **any entity that provides an interface**:

- **Language Virtualization**
- **Process Virtualization**
- **System Virtualization**

A Virtual “Machine”

★ A “**machine**” is: **any entity that provides an interface**:

- **Language Virtualization**
  - Machine := Entity that provides the API
- **Process Virtualization**
  - Machine := Entity that provides the ABI
- **System Virtualization**
  - Machine := Entity that provides the ISA

Language Virtualization

Language Virtualization Example

Initial State (S_1):		
Transition ($S_1 \Rightarrow S_2$):		

System #1
`COPY r1 1`
`SHIFTL x 2`
`ADD x r1`
	System #2
	`COPY r1 x`
	`SHIFTL x`
	`SHIFTL x`
	`ADD x r1`
		System #3
		`COPY r1 x`
		`ADD r1 x`
		`ADD r1 x`
		`ADD r1 x`
		`ADD r1 x`

Final State (S_2):		

Process Virtualization

Process Virtualization Example

Initial State ($\mathbf{S_1}$):
Transition ($\mathbf{S_1} \Rightarrow \mathbf{S_2}$):
Final State ($\mathbf{S_2}$):

System Virtualization

System VMs

- ★ **Type 1 Hypervisor**

System VMs

★ **Type 1 Hypervisor**

- Implement **on bare hardware**

- Most efficient,
- Must support hardware emulation (drivers), and
- Replaces any OS hosted on the bare hardware.

System VMs

- ★ **Type 2 Hypervisor**

System VMs

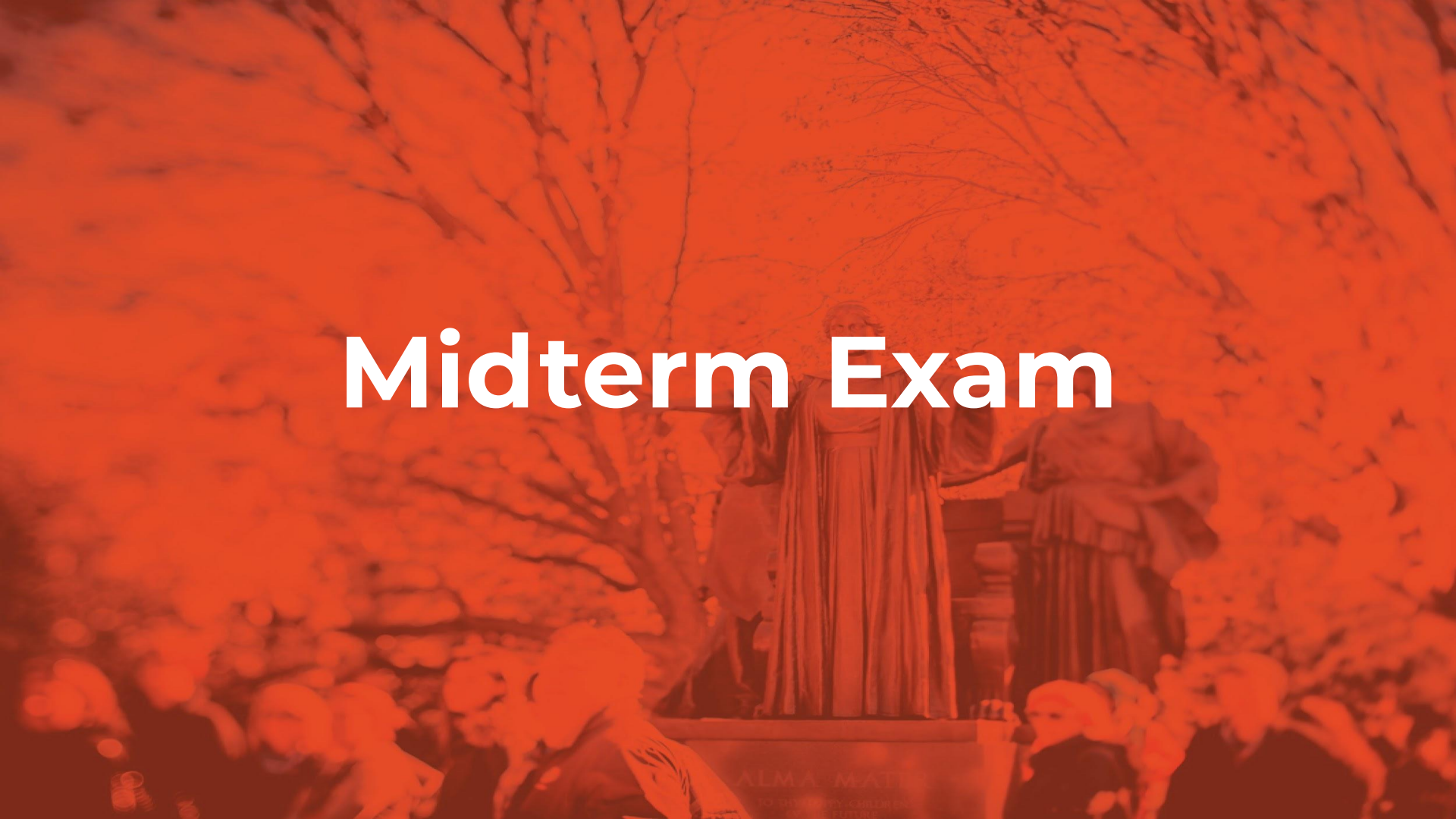
★ **Type 2 Hypervisor**

- Implement a VMM **on top of a host OS**:

- Less efficient,
- Leverages the OS drivers and hardware abstractions, and
- Easy to install on top of the host OS.

System VMs

- ★ **How has this changed our industry?**



Midterm Exam

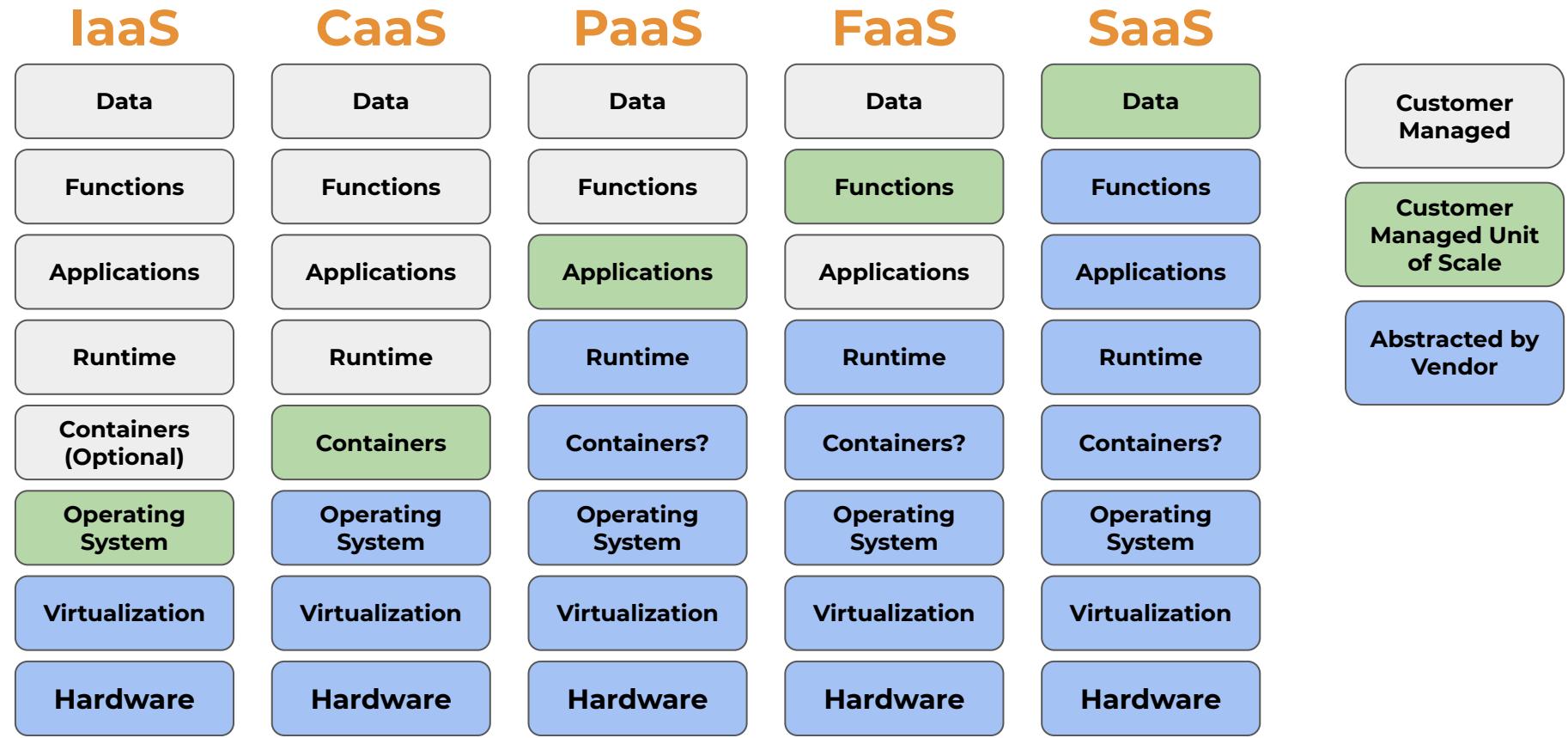
Cloud-Scale Abstractions

IaaS	CaaS	PaaS	FaaS	SaaS
Data	Data	Data	Data	Data
Functions	Functions	Functions	Functions	Functions
Applications	Applications	Applications	Applications	Applications
Runtime	Runtime	Runtime	Runtime	Runtime
Containers (Optional)	Containers	Containers?	Containers?	Containers?
Operating System	Operating System	Operating System	Operating System	Operating System
Virtualization	Virtualization	Virtualization	Virtualization	Virtualization
Hardware	Hardware	Hardware	Hardware	Hardware

Customer Managed

Customer Managed Unit of Scale

Abstracted by Vendor


IaaS
CaaS
PaaS
FaaS
SaaS
Data
Functions
Applications
Runtime
Containers (Optional)
Containers
Containers?
Operating System
Virtualization
Hardware
Customer Managed
Customer Managed Unit of Scale
Abstracted by Vendor

Infrastructure as a Service (IaaS)

Containers as a Service (CaaS)

Containers provide an _____ of a system that can be deployed in an isolated environment on heterogeneous systems.

Key Technology:

Docker Containers

Container Developer:

Container User:

```
1 FROM gcc:latest
2 COPY ./docker/entrypoint.sh /
3 RUN chmod +x entrypoint.sh
4 ENTRYPOINT ["/entrypoint.sh"]
```

```
docker build --tag mp3-docker .
```

```
docker run -it --rm -v "pwd":/mp3
                            mp3-docker "make"
```

```
docker run --rm -it -p 27017:27017 mongo
```