

Distributed Systems

CS6421

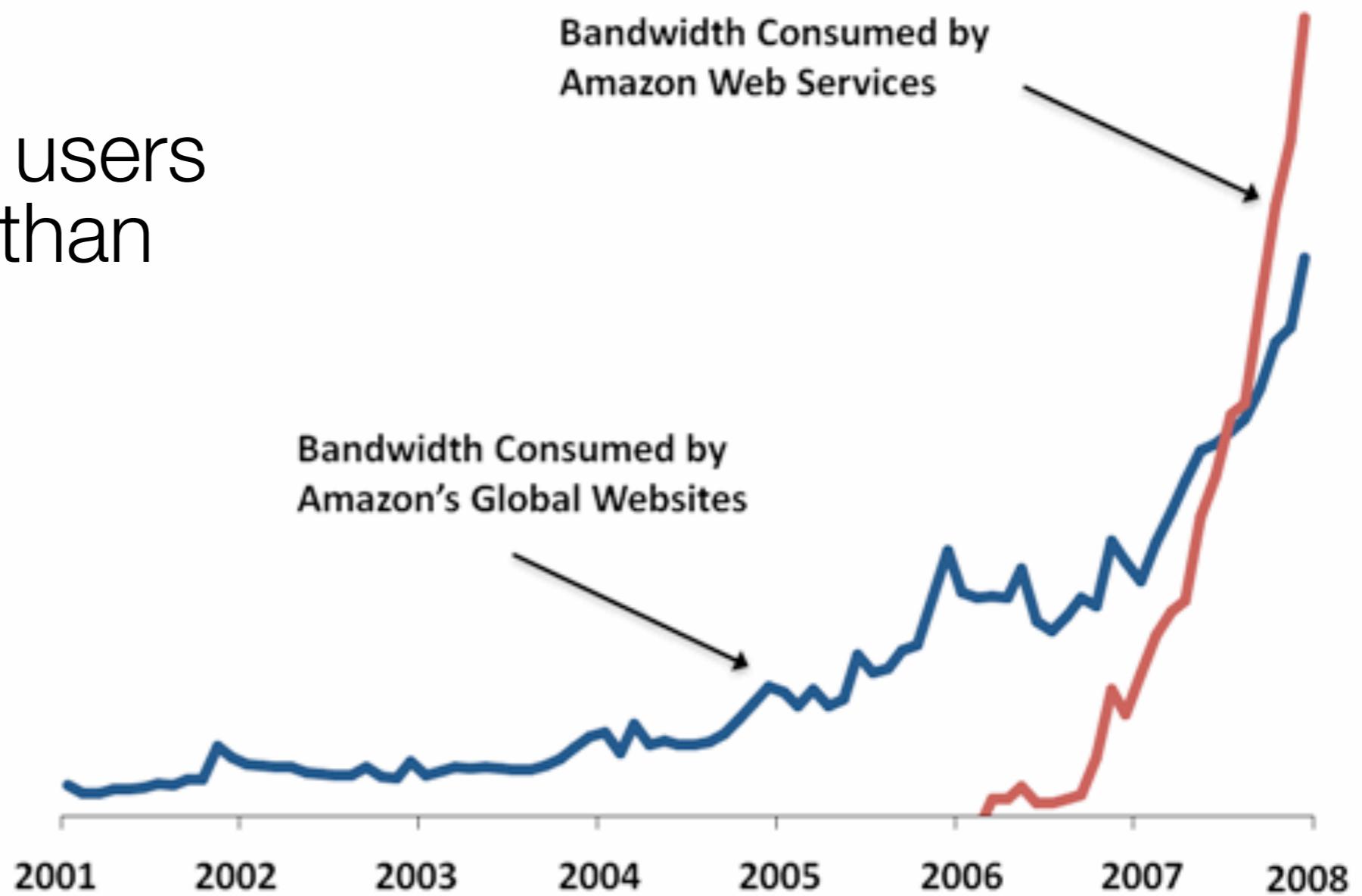
Cloud Computing: Servers and Virtualization

Prof. Tim Wood

Amazon's Cloud

Amazon built its cloud platform so that other people could pay for its infrastructure during the rest of the year...

Now its cloud users are far bigger than its own sites



Cloud Data Centers



Microsoft's Dublin
data center

Interconnections

Amazon's Internet

- Multiple private 100Gbps links between each data center site



Servers in AWS

Custom server designs

1U compute servers

- Intel CPUs
- High efficiency power supplies

Storage Racks

- 42U size
- 1100 disks
- 11 petabytes of storage space



Scale Estimates

- 1.5-2 million servers - Bloomberg 2014
- 50-80K per data center, 68 total data centers = 3.4-5.4 million
 - re:Invent 2016

*Every day Amazon adds as many servers as it had in **2000** (when it was a **\$2 billion** company)*
— talk at UW **2011**

*Every day Amazon adds as many servers as it had in **2005** (when it was a **\$8.5 billion** company)*
— AWS re:Invent **2016**

Inside a Data Center



<https://www.google.com/about/datacenters/inside/streetview/>

Why use the cloud?

- Pay-as-you go
- Expand quickly on demand
- Don't need to worry about (many) IT issues
- Cheap!

... but is the cloud perfect?

[spoiler alert] no.

Infrastructure as a Service (IaaS)

Infrastructure clouds rent **raw servers**

- Connect to server remotely
- Configure OS and install whatever applications you want

Great flexibility for cloud user

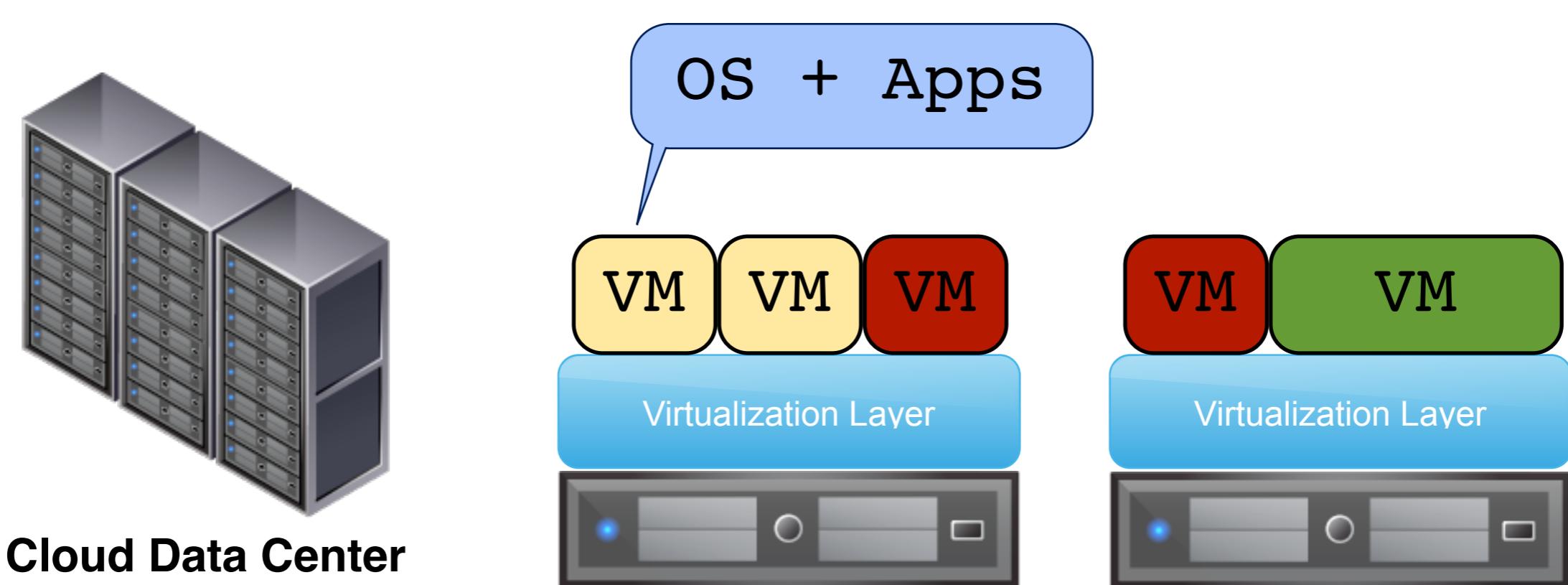
Less management handled by cloud operator

**Your own computer or disk
on demand !**

Virtual Machines

Virtualization is used to **split up** a physical server

- Allows multiple customers to share one machine
- Simplifies management since VMs are not strictly tied to HW
- Provides isolation between cloud users



Amazon EC2

- Infrastructure as a Service Cloud (IaaS)
- Can rent server and storage resources

	Description	Cost
t3.Micro	1GB RAM, up to 1 core, no storage	\$0.01 / hour
t3.Large	8GB RAM, ~2 cores, no storage	\$0.08 / hour
c5.18xlarge	144GB RAM, 72 cores, no storage	\$3.06 / hour

EBS	Network attached storage	\$0.10 / GB per month
------------	--------------------------	-----------------------

Platform as a Service (PaaS)

The cloud provides a **programming platform**

Typically used to run highly scalable web apps

Cloud users write applications to run on the cloud

- Must write code to meet cloud API
- Cloud automatically scales the application based on demand
- Provides much greater scalability, but program must be specially written

**Let the cloud handle your
application's scalability!**

Software as a Service (SaaS)

The cloud provides a **piece of software**

- Examples: email, office, project management, customer relations, supply chain, etc

Provides even greater scalability

- Entire cloud infrastructure is devoted just to one particular type of application

Benefits for customer: cheaper and simpler

Benefits for provider: economy of scale

**Why bother writing or running
your own application if they can
do it better?**

Examples

PaaS

- Google App Engine
 - Python, Java
- Heroku
 - Ruby on Rails
- Amazon EMR
 - Java, Python, etc Hadoop

SaaS

- GMail
- Flickr
- Salesforce
- Dropbox
- iCloud

Cloud Grade Sheet

	Pay as you go	Scalability	Automation / ease of use	Flexibility	Security / Isolation
IaaS	++	+	-	++	++
PaaS	+++	++	++	- +	-
SaaS	+++	+++	+++	- - -	- +
Private Data Center	- - -	-	- -	++ +	+++++ +++ +

Types of Clouds

Software as a Service



Salesforce

Office apps, CRM

for anybody

Platform as a Service



Google
App Engine™

heroku

Software platforms

for programmers

Infrastructure as a Service



amazon
web services™



Azure

Servers & storage

for programmers
and sys admins

Increased
Cloud
Automation

Increased
Customer
Control



Cloud Computing Goals

Offer fast services to customers worldwide

- Need geographic diversity and high scalability
- Low latency requests: fast responses
- High throughput: simultaneous processing

Ads ⓘ

[Google Cloud Computing](#)
www.google.com/apps/business
Save time & money with Google Apps for Business. 30 days free!

...that are highly reliable and secure

- Servers crash
- Data centers lose power
- Malicious users (or governments?) can attack

[IBM Cloud Computing](#)
www.ibm.com/cloud
Reinvent Business Processes & Drive Innovation. Explore IBM Solutions.
175 people +1'd this page

... as cheaply as possible

- Users expect services for free*
- Cloud needs to pay for servers, cooling infrastructure, energy, system administrators, etc

[Top 5 Truths of the Cloud](#)
www.citrix.com/
Learn the Essentials with Citrix.
Download the Free Whitepaper Now.

Let's try out the cloud



AWS in 2012...

Welcome

The AWS Management Console provides a graphical interface to Amazon Web Services. Learn more about how to use our services to meet your needs, or get started by selecting a service.

[Getting started guides](#)

[Reference architectures](#)

[Free Usage Tier](#)

Set Start Page

[Console Home](#)

Amazon Web Services

Compute & Networking

-  **EC2**
Virtual Servers in the Cloud
-  **Elastic MapReduce**
Managed Hadoop Framework
-  **Route 53**
Scalable Domain Name System
-  **VPC**
Isolated Cloud Resources

Storage & Content Delivery

-  **CloudFront**
Global Content Delivery Network
-  **S3**
Scalable Storage in the Cloud
-  **Storage Gateway**
Integrates on-premises IT environments with Cloud storage

Database

-  **DynamoDB**
Predictable and Scalable NoSQL Data Store
-  **ElastiCache**
In-Memory Cache
-  **RDS**
Managed Relational Database Service

Deployment & Management

-  **CloudFormation**
Templated AWS Resource Creation
-  **CloudWatch**
Resource & Application Monitoring
-  **Elastic Beanstalk**
AWS Application Container
-  **IAM**
Secure AWS Access Control

App Services

-  **CloudSearch**
Managed Search Service
-  **SES**
Email Sending Service
-  **SNS**
Push Notification Service
-  **SQS**
Message Queue Service
-  **SWF**
Workflow Service for Coordinating Application Components

Announcements

[Announcing VM Export for Amazon EC2](#)

[AWS Console Enhancements for Elastic Load Balancing: Listener, Certificate, and...](#)

[Amazon RDS announces support for MySQL Read Replica in Amazon VPC](#)

[More...](#)

Service Health [Edit](#)

Click Edit to add at least one service and at least one region to monitor.

[Service Health Dashboard](#)

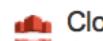
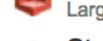
AWS in 2015...

Amazon Web Services

Compute

-  **EC2**
Virtual Servers in the Cloud
-  **EC2 Container Service**
Run and Manage Docker Containers
-  **Elastic Beanstalk**
Run and Manage Web Apps
-  **Lambda**
Run Code in Response to Events

Storage & Content Delivery

-  **S3**
Scalable Storage in the Cloud
-  **CloudFront**
Global Content Delivery Network
-  **Elastic File System PREVIEW**
Fully Managed File System for EC2
-  **Glacier**
Archive Storage in the Cloud
-  **Import/Export Snowball**
Large Scale Data Transport
-  **Storage Gateway**
Integrates On-Premises IT Environments with Cloud Storage

Database

-  **RDS**
Managed Relational Database Service
-  **DynamoDB**
Predictable and Scalable NoSQL Data Store
-  **ElastiCache**
In-Memory Cache
-  **Redshift**
Managed Petabyte-Scale Data Warehouse Service

Networking

-  **VPC**
Isolated Cloud Resources
-  **Direct Connect**
Dedicated Network Connection to AWS
-  **Route 53**
Scalable DNS and Domain Name Registration

Developer Tools

-  **CodeCommit**
Store Code in Private Git Repositories
-  **CodeDeploy**
Automate Code Deployments
-  **CodePipeline**
Release Software using Continuous Delivery

Management Tools

-  **CloudWatch**
Monitor Resources and Applications
-  **CloudFormation**
Create and Manage Resources with Templates
-  **CloudTrail**
Track User Activity and API Usage
-  **Config**
Track Resource Inventory and Changes
-  **OpsWorks**
Automate Operations with Chef
-  **Service Catalog**
Create and Use Standardized Products
-  **Trusted Advisor**
Optimize Performance and Security

Security & Identity

-  **Identity & Access Management**
Manage User Access and Encryption Keys
-  **Directory Service**
Host and Manage Active Directory
-  **Inspector PREVIEW**
Analyze Application Security
-  **WAF**
Filter Malicious Web Traffic

Analytics

-  **EMR**
Managed Hadoop Framework
-  **Data Pipeline**
Orchestration for Data-Driven Workflows
-  **Elasticsearch Service**
Run and Scale Elasticsearch Clusters
-  **Kinesis**
Work with Real-time Streaming data
-  **Machine Learning**
Build Smart Applications Quickly and Easily

Internet of Things

-  **AWS IoT BETA**
Connect Devices to the cloud

Mobile Services

-  **Mobile Hub BETA**
Build, Test, and Monitor Mobile apps
-  **Cognito**
User Identity and App Data Synchronization
-  **Device Farm**
Test Android, Fire OS, and iOS apps on real devices in the Cloud
-  **Mobile Analytics**
Collect, View and Export App Analytics
-  **SNS**
Push Notification Service

Application Services

-  **API Gateway**
Build, Deploy and Manage APIs
-  **AppStream**
Low Latency Application Streaming
-  **CloudSearch**
Managed Search Service
-  **Elastic Transcoder**
Easy-to-use Scalable Media Transcoding
-  **SES**
Email Sending Service
-  **SQS**
Message Queue Service
-  **SWF**
Workflow Service for Coordinating Application Components

Enterprise Applications

-  **WorkSpaces**
Desktops in the Cloud
-  **WorkDocs**
Secure Enterprise Storage and Sharing Service
-  **WorkMail PREVIEW**
Secure Email and Calendaring Service

AWS in 2018!

AWS services

Find a service by name or feature (for example, EC2, S3 or VM, storage).

> Recently visited services

▽ All services

Compute

- EC2
- Lightsail ↗
- Elastic Container Service
- EKS
- Lambda
- Batch
- Elastic Beanstalk

Storage

- S3
- EFS
- Glacier
- Storage Gateway

Database

- RDS
- DynamoDB
- ElastiCache
- Neptune
- Amazon Redshift

Migration

- AWS Migration Hub
- Application Discovery Service
- Database Migration Service
- Server Migration Service
- Snowball

Networking & Content Delivery

- VPC
- CloudFront
- Route 53
- API Gateway
- Direct Connect

Developer Tools

- CodeStar
- CodeCommit
- CodeBuild
- CodeDeploy
- CodePipeline
- Cloud9
- X-Ray

Management Tools

- CloudWatch
- AWS Auto Scaling
- CloudFormation
- CloudTrail
- Config
- OpsWorks
- Service Catalog
- Systems Manager
- Trusted Advisor
- Managed Services

Mobile Services

- Mobile Hub
- AWS AppSync
- Device Farm
- Mobile Analytics

AR & VR

- Amazon Sumerian

Application Integration

- Step Functions
- Amazon MQ
- Simple Notification Service
- Simple Queue Service
- SWF

Customer Engagement

- Amazon Connect
- Pinpoint
- Simple Email Service

Machine Learning

- Amazon SageMaker
- Amazon Comprehend
- AWS DeepLens
- Amazon Lex
- Machine Learning
- Amazon Polly
- Rekognition
- Amazon Transcribe
- Amazon Translate

Business Productivity

- Alexa for Business
- Amazon Chime ↗
- WorkDocs
- WorkMail

Desktop & App Streaming

- WorkSpaces
- AppStream 2.0

Internet of Things

- IoT Core
- IoT 1-Click
- IoT Device Management
- IoT Analytics
- Greengrass
- Amazon FreeRTOS
- IoT Device Defender

Security, Identity & Compliance

- IAM
- Cognito
- Secrets Manager
- GuardDuty
- Inspector
- Amazon Macie ↗
- AWS Single Sign-On
- Certificate Manager
- CloudHSM
- Directory Service
- WAF & Shield
- Artifact

Game Development

- Amazon GameLift

Let's try it out!

<https://console.aws.amazon.com>

Instance details:

- Auto-assign Public IP -> Enable

```
sudo apt-get update
```

```
sudo apt-get install -y sysbench
```

```
sysbench --test=cpu --num-threads=100 --max-requests=50000 run
```

Why is it so cheap?

c5.18xlarge - **\$3.06 per hour**

- 144GB RAM, 72x ~3Ghz CPU cores, 25Gbps

If busy 24x365 = \$26,805.60 per year!

Could just buy from Dell...

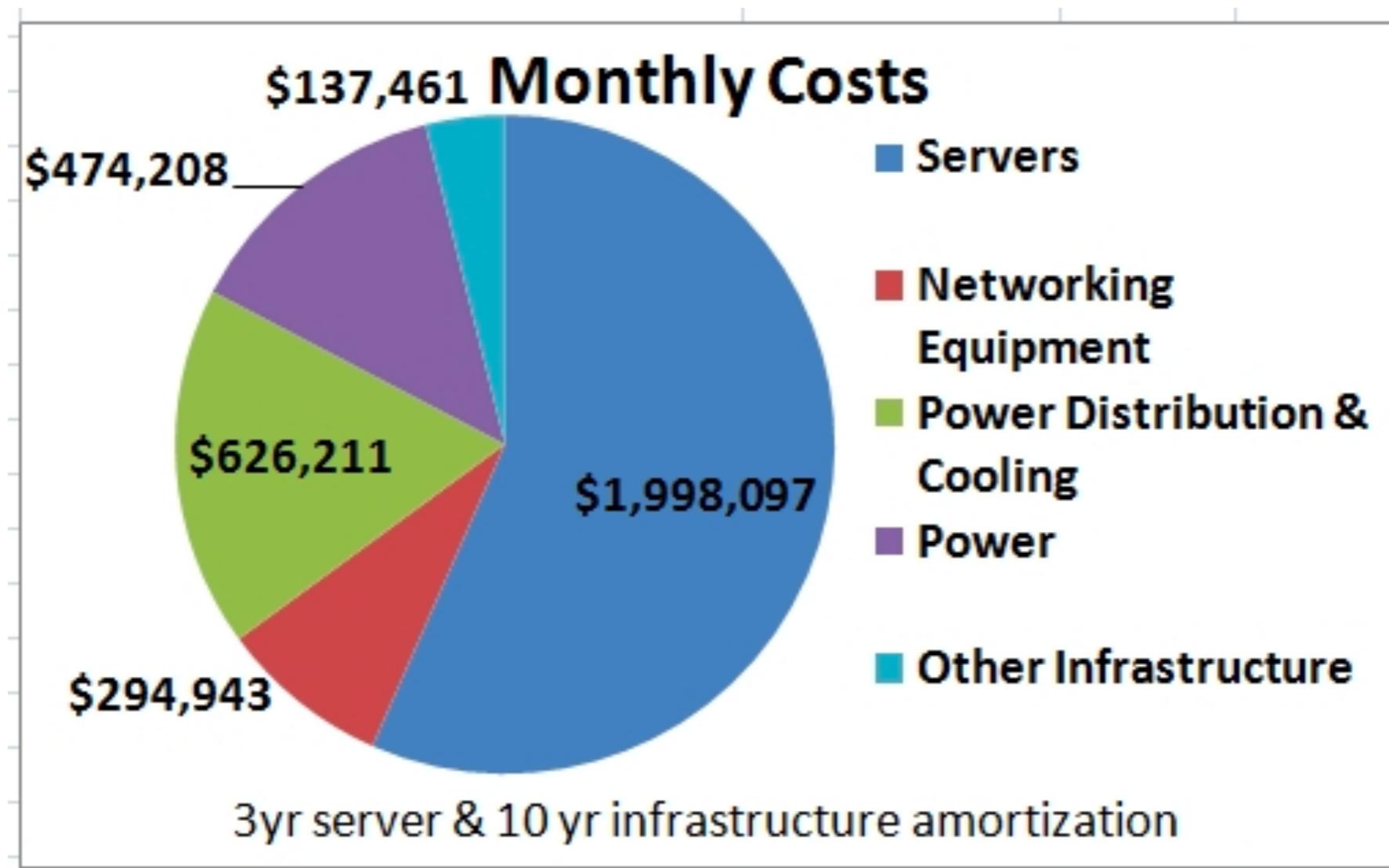
PowerEdge R930 Rack Server Summary

Dell Price	\$37,245.40
Starting at Price	\$59,210.00
Total Savings	\$21,964.60
Standard Delivery	Free



It's not cheap

The cost to run a 50,000 server data center (2010):



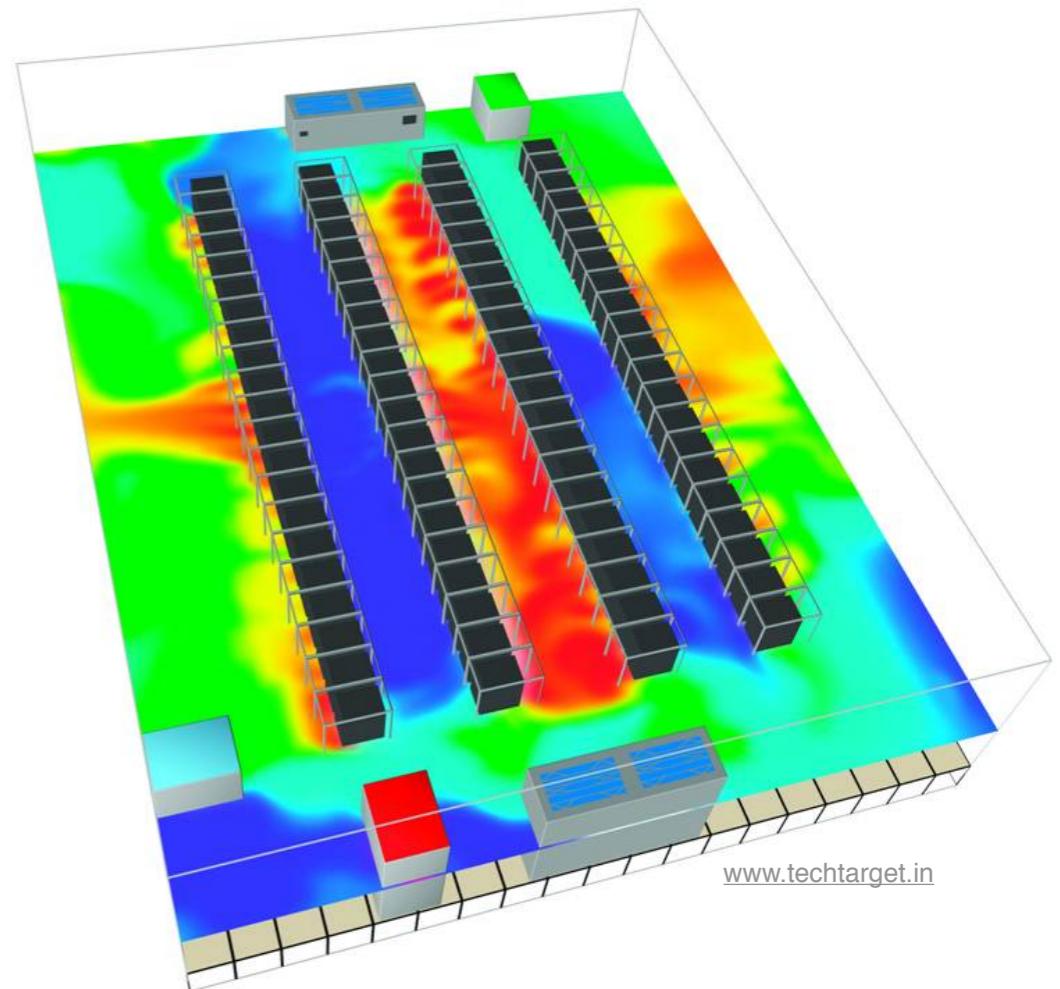
Heat and Power

Computers are hot!

- Thermostat set to 55-72 degrees
- Hot and cold air aisles
- Infrared mapping to find hotspots
- Complex thermodynamic models
- A Finnish data center pipes the heat to warm 1,000 nearby homes

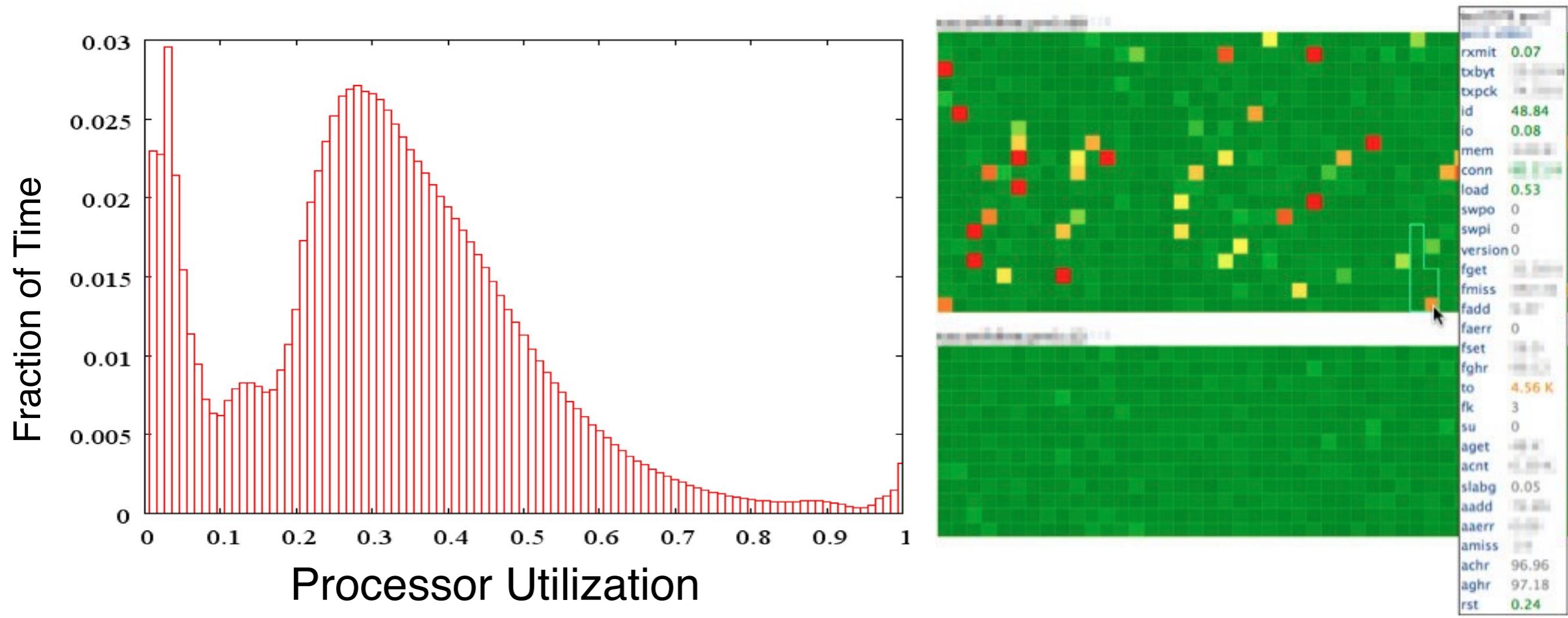
Computers use power!

- Several megawatts per data center
- 1.3% of world electricity usage
- Often, only 50% of a data center's energy goes to actual IT equipment



Server (In)Efficiency

Many servers are poorly utilized



How can we improve this?

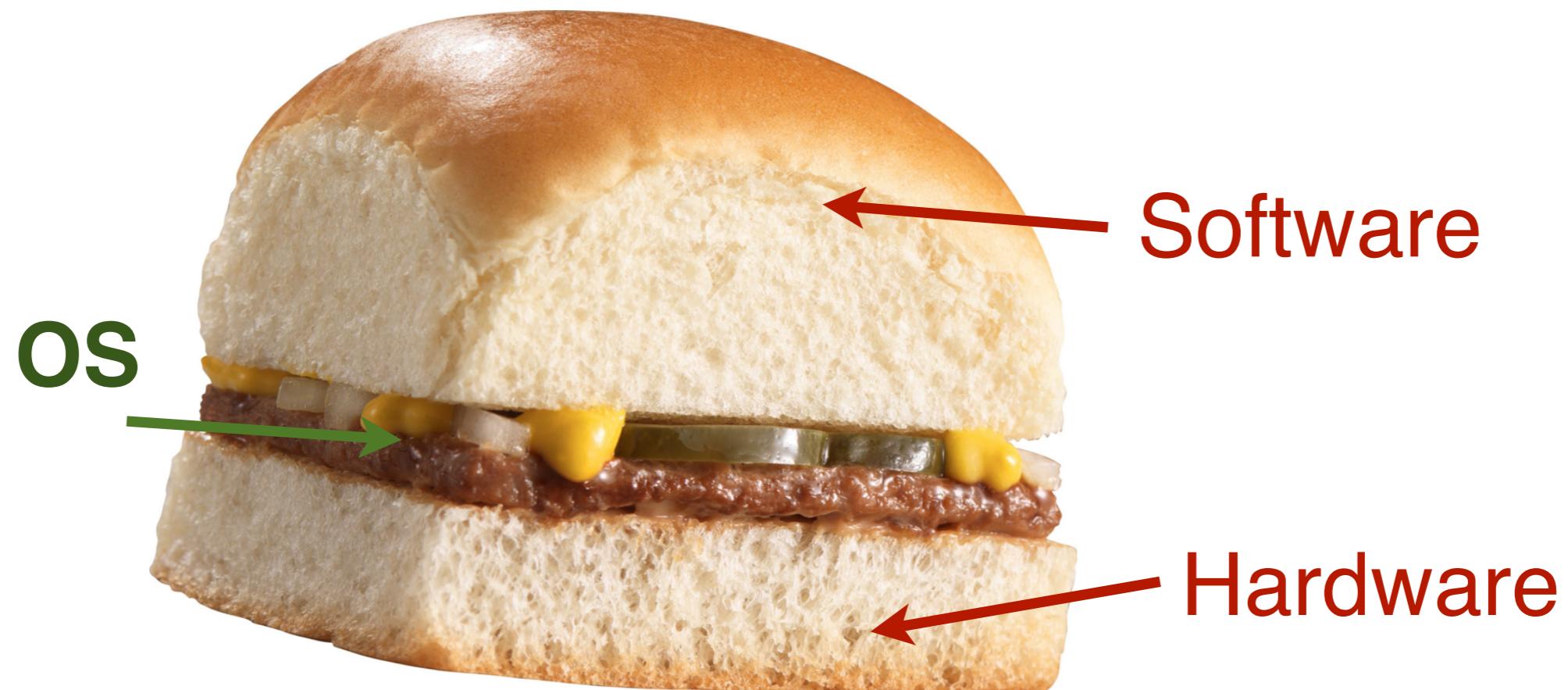
Figure from: The Data Center as a Computer by Luiz André Barroso and Urs Hözle

Break?

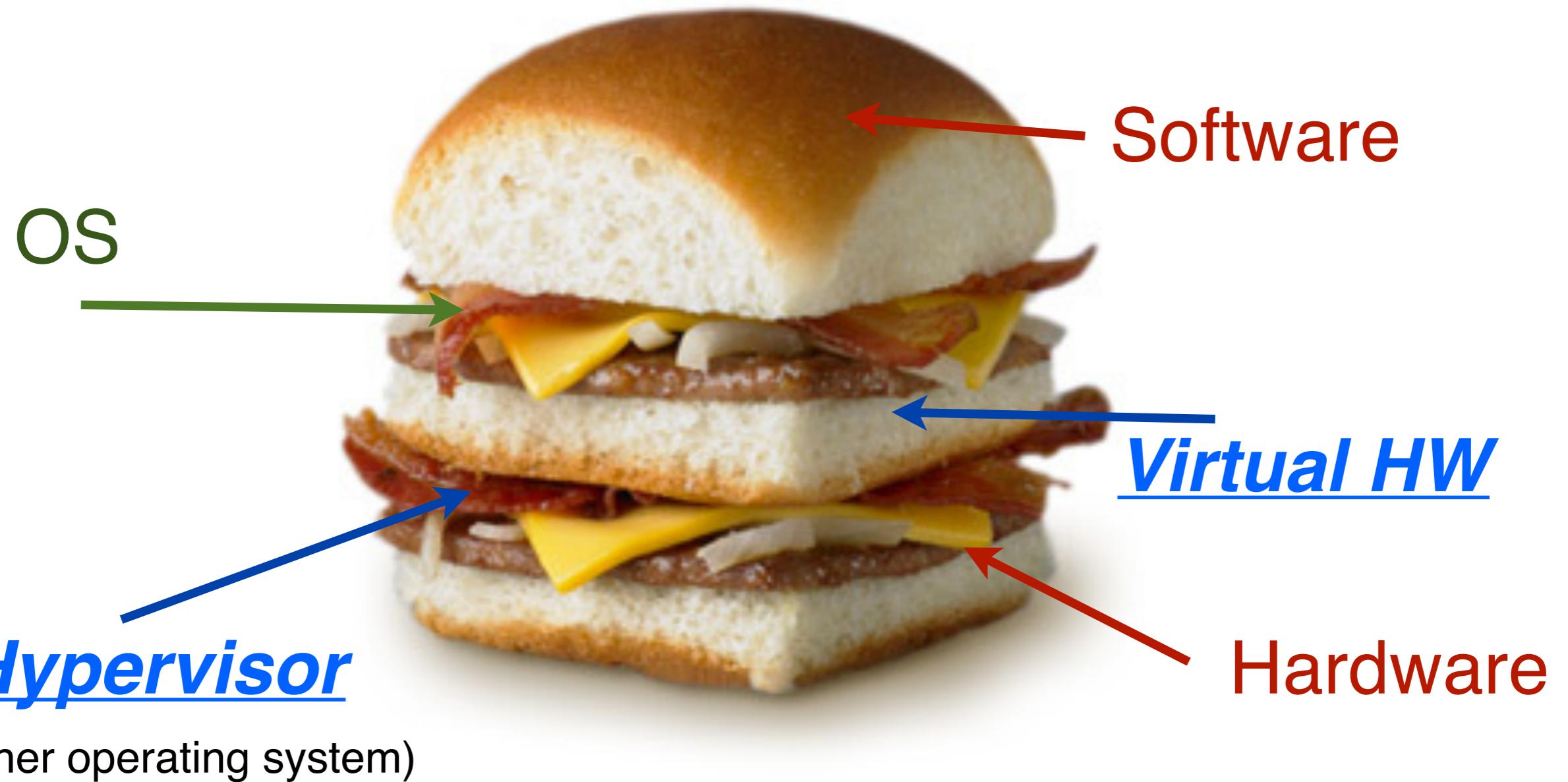
Virtualizing Resources

Question...

What's better than an operating system?



Virtualization



Multiple VMs

Hypervisor can manage many virtual machines

- Just like OS manages many processes

Windows
desktop VM

Linux web
server VM

Obscure-OS
running ??? VM



Hypervisor

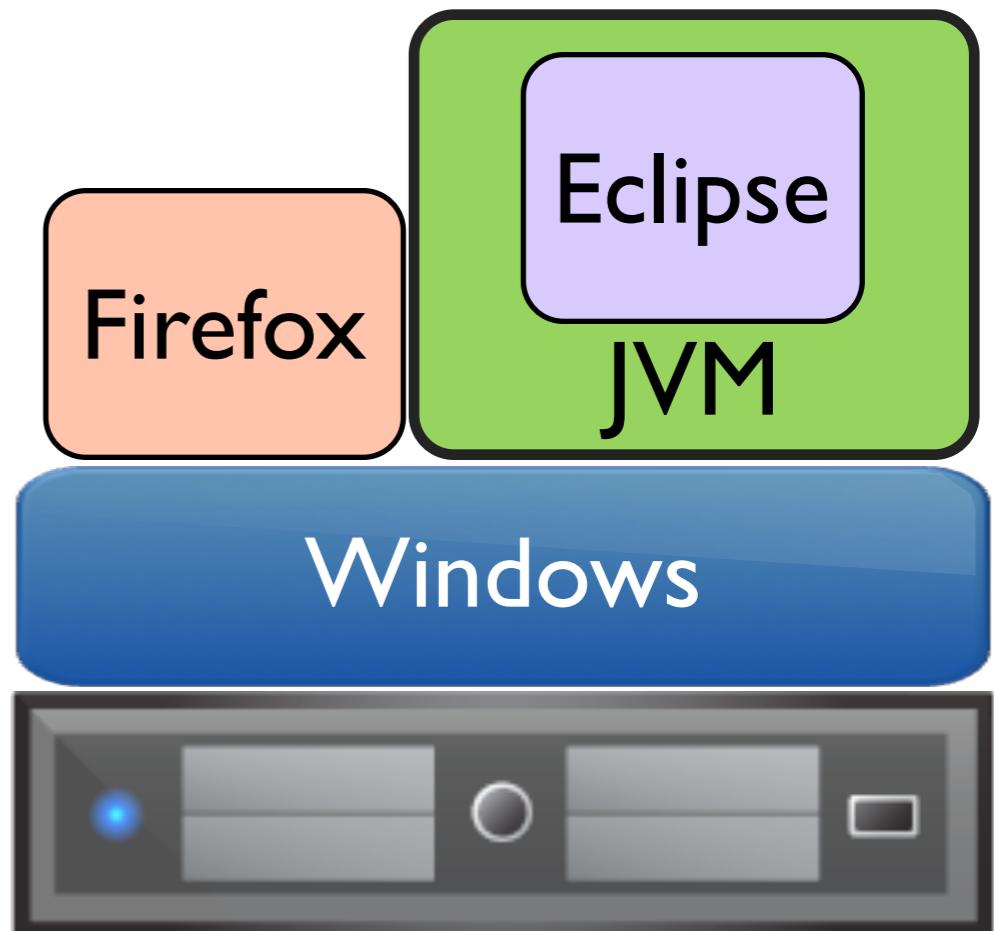
The Most Common VM

Java Virtual Machine

- Execution environment for running Java code
- Interprets/compiles programs and translates them to the host environment

Abstraction layer to OS

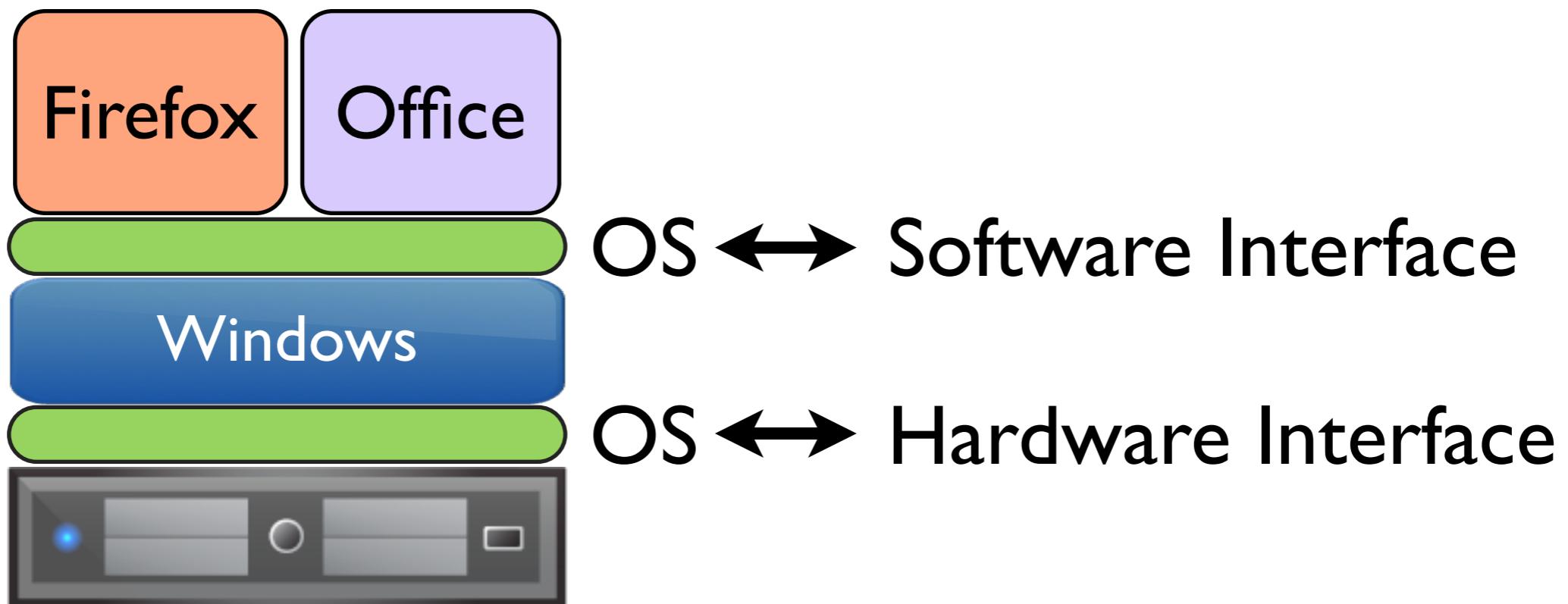
- Java code is platform independent
- Can add overhead



What is virtualization?

An extra **interface** that mimics the behavior of a lower layer

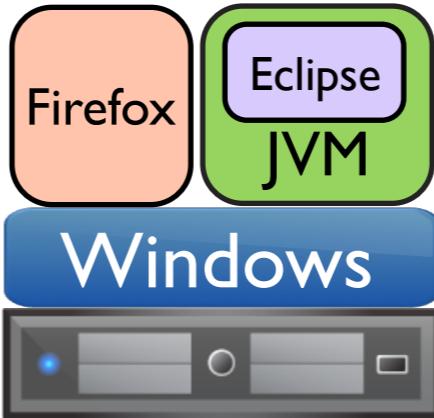
Used since 1970s so new mainframes could support legacy applications



Types of Virtualization

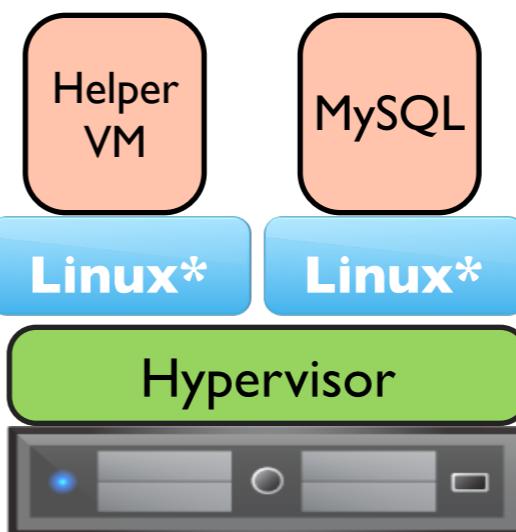
Application Virtualization

- Runs application code
- *Java JVM, WINE*



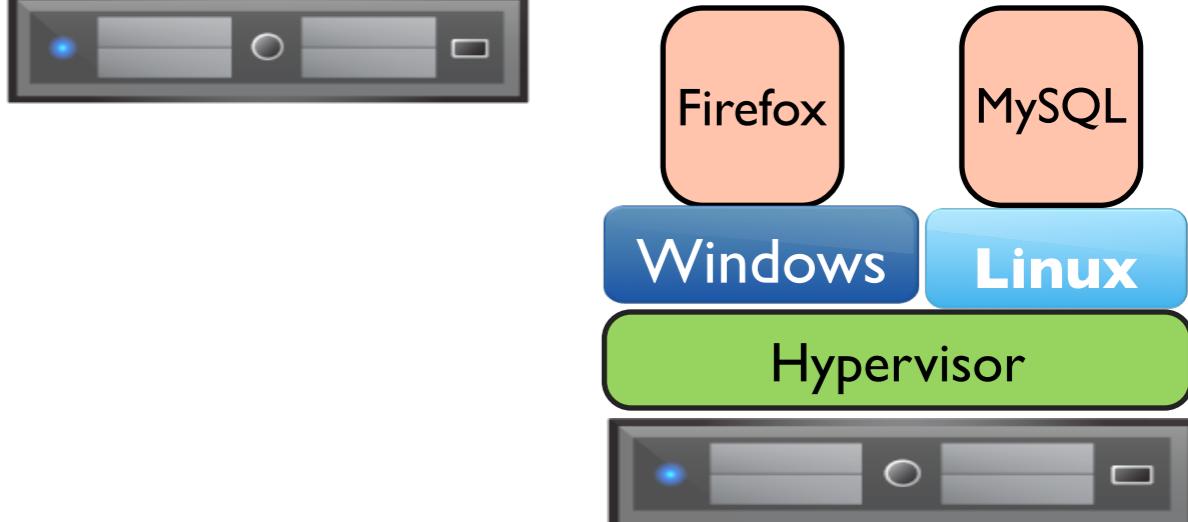
Hosted Virtualization

- Virtualizes a full OS and apps
- *VMware Player, VirtualBox*



Paravirtualization

- Modify OS to simplify hypervisor
- *Xen*



Full Virtualization

- Runs directly on HW
- *VMware ESXi*

Why Virtualize?

Consolidation

- Can split a physical server into many smaller servers

Security

- VMs are isolated from one another

Resource management

- Can dynamically adjust a VM's CPU and memory share

Convenience

- VM is abstracted away from physical hardware
- Great for development

How to Virtualize?

Virtualization layer replaces an interface

Must intercept calls and translate them

- Java - interpret/compile code to match host
- Hosted VM - translate system calls for host OS
- Full Virtualization - trap on sensitive instructions

How to allocate resources?

- VMs must share memory and CPU time

How to handle I/O?

- Abstraction layer separates VM from physical hardware

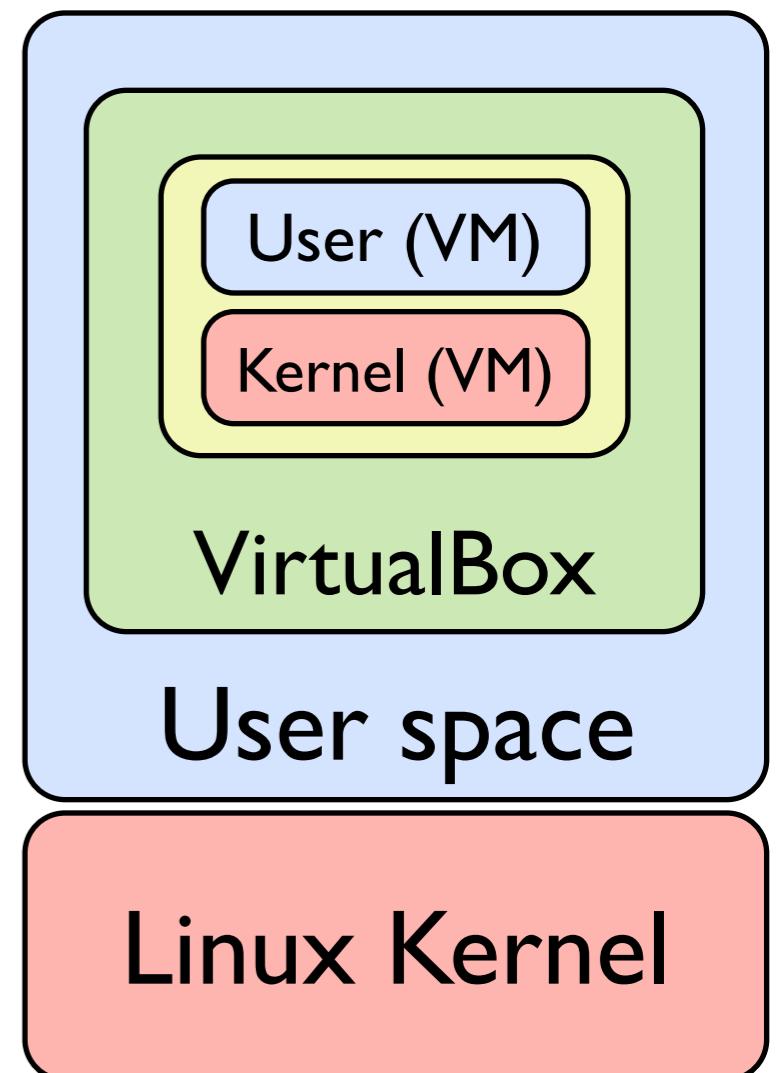
Hosted Virtualization

Normal OS divided into Kernel and User modes

Protected instructions only work in kernel mode

- I/O, memory allocation, etc
- Traps to kernel if run in user mode

How to run a VM in user mode?

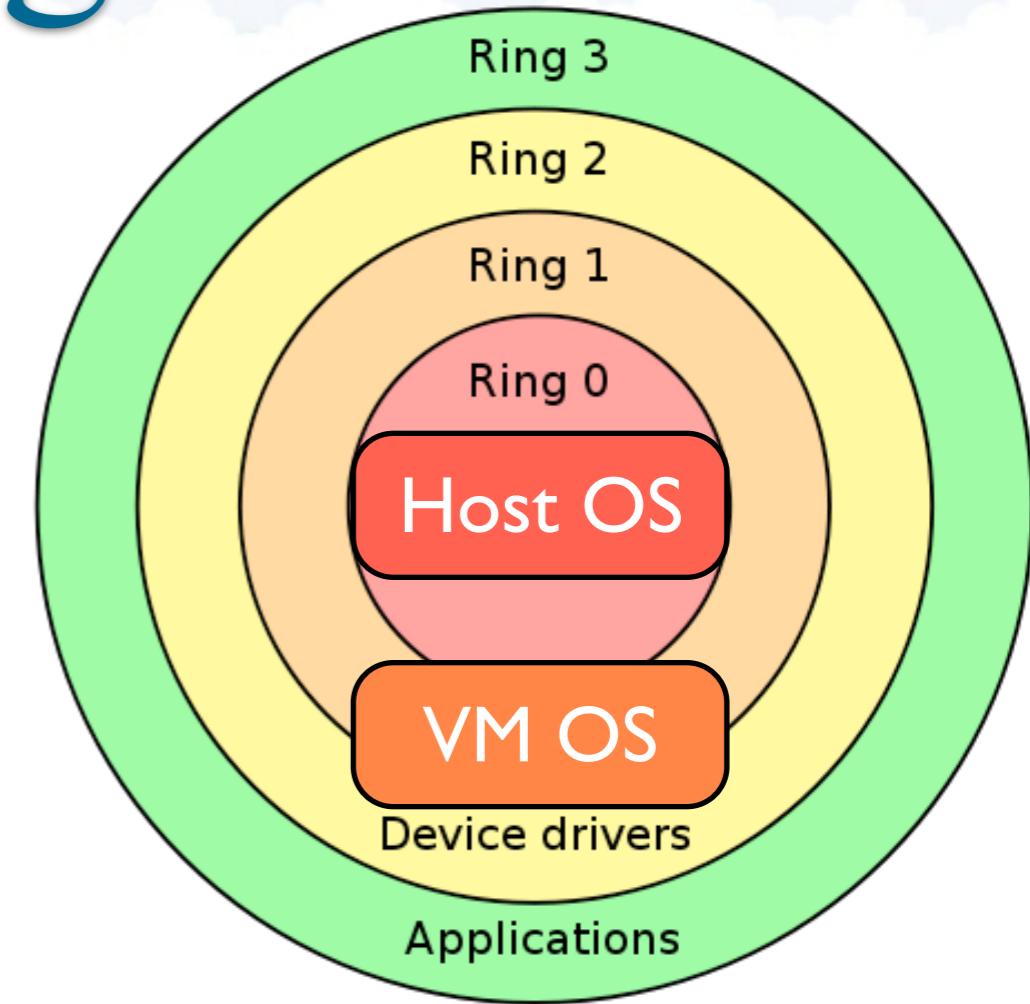


CPU Rings

User and kernel mode are controlled by CPU

Modern CPUs support multiple protection rings

- Ring 0 = kernel mode
- Ring 3 = user mode
- Rings 1-2 = drivers or unused



Hosted virtualization runs VM OS in Ring 1

- Must detect and translate any CPU instructions that require Ring 0

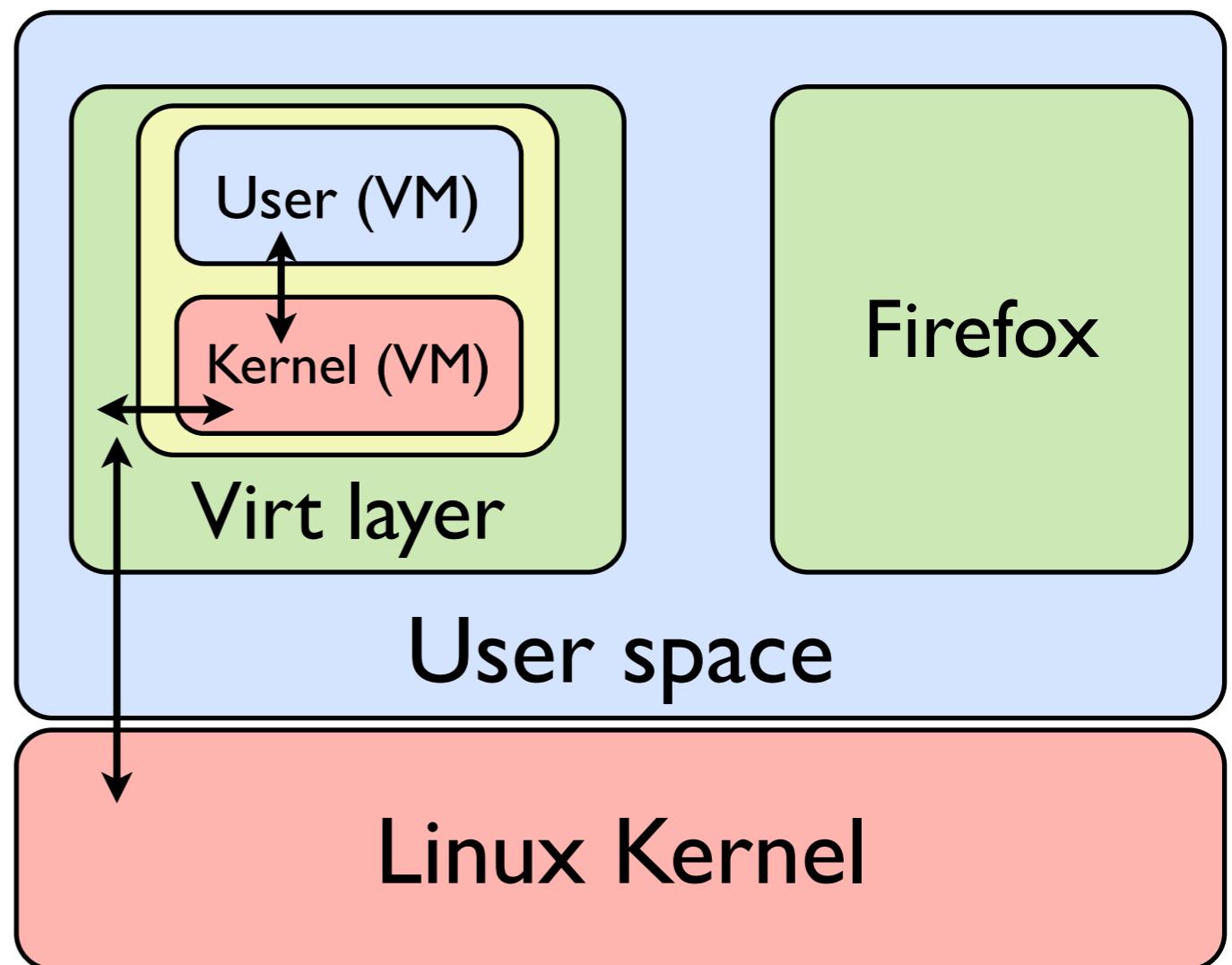
Ring 0 ops
set time
power on/off
memory management
etc

Hosted Virtualization

Dynamic translation

- Preprocess all code being run by the OS inside the VM
- Detect sensitive instructions
- Repackage and call into parent OS
- Return result to guest OS

How to optimize?



Full Virtualization

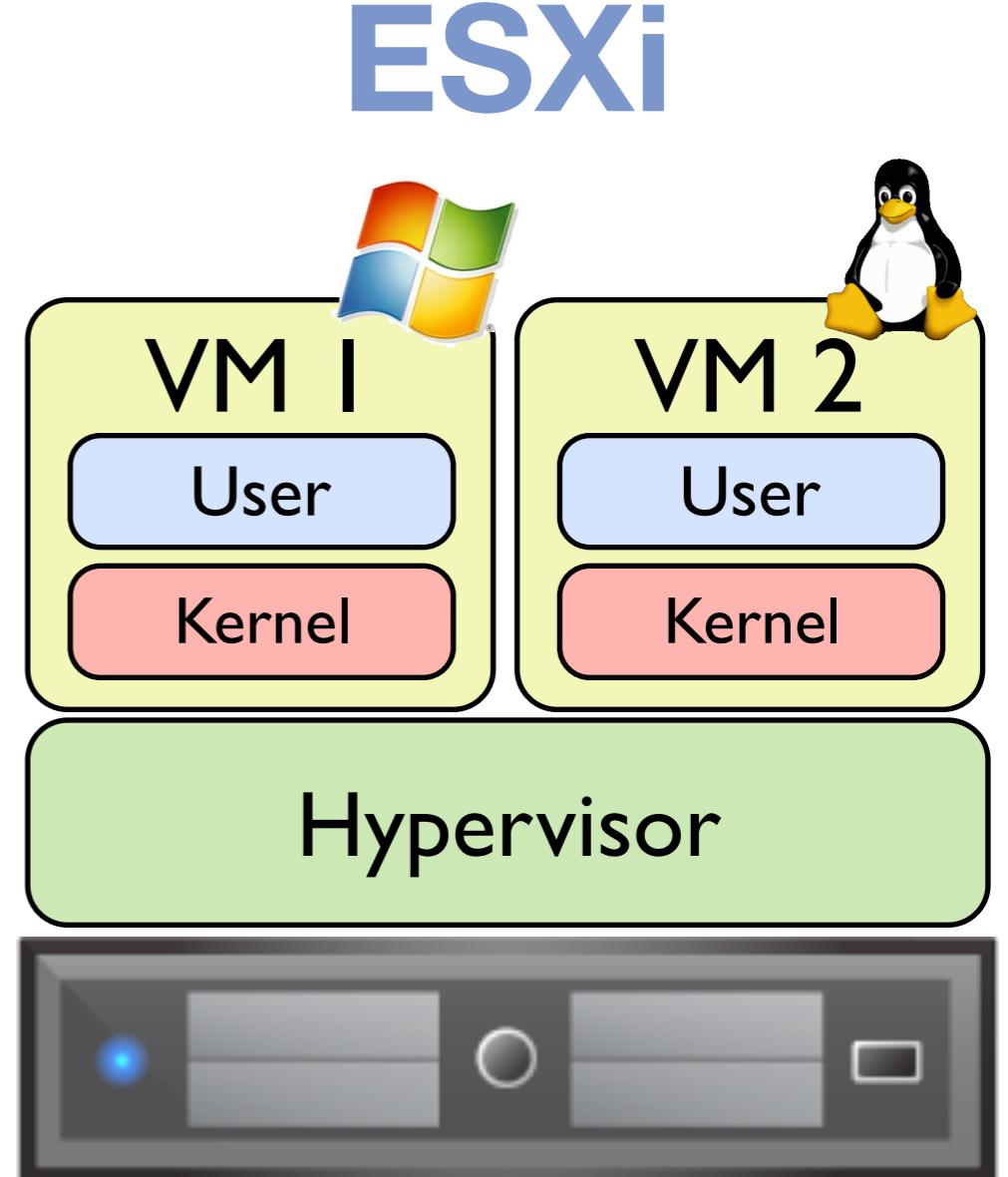
Hypervisor runs directly on hardware in Ring 0



Manages VMs

Uses dynamic translation to rewrite protected instructions

Hosts device drivers for VMs



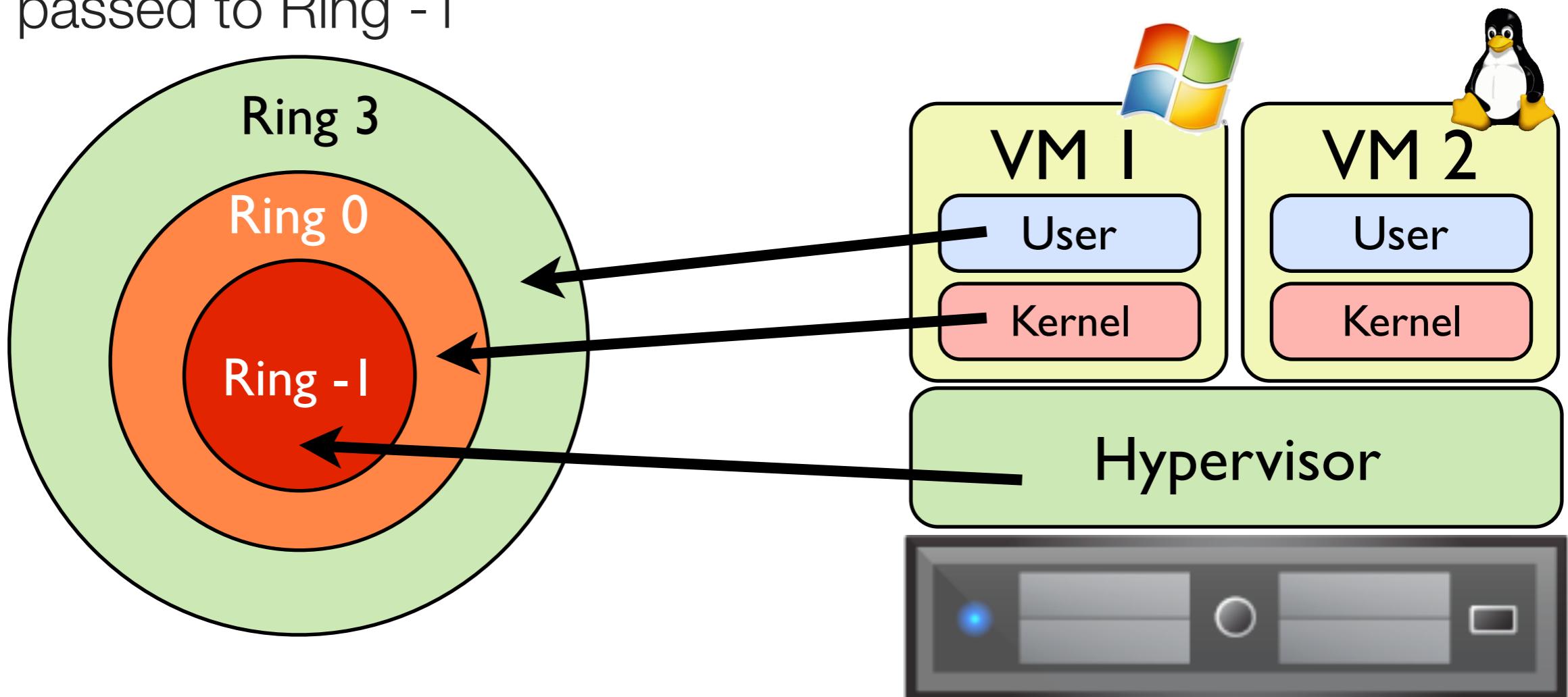
HW Virtualization

Newer CPUs have support for virtualization

- AMD-V and Intel-VT

Provides an extra ring for running a hypervisor

- Protected instructions in VM OS are trapped and passed to Ring -1



Getting help from the VM

Hosted and Full virtualization are VM OS agnostic

- Guest OS does not know it is being virtualized
- Translate binary code (slow)
- Get help from hardware (expensive)

What if we ask the VM's OS for help?

- Have guest OS notify hypervisor of special instructions
- Guest OS can help with device drivers

Benefits and drawbacks?

Paravirtualization in



Modifies Linux so that it is virtualization aware

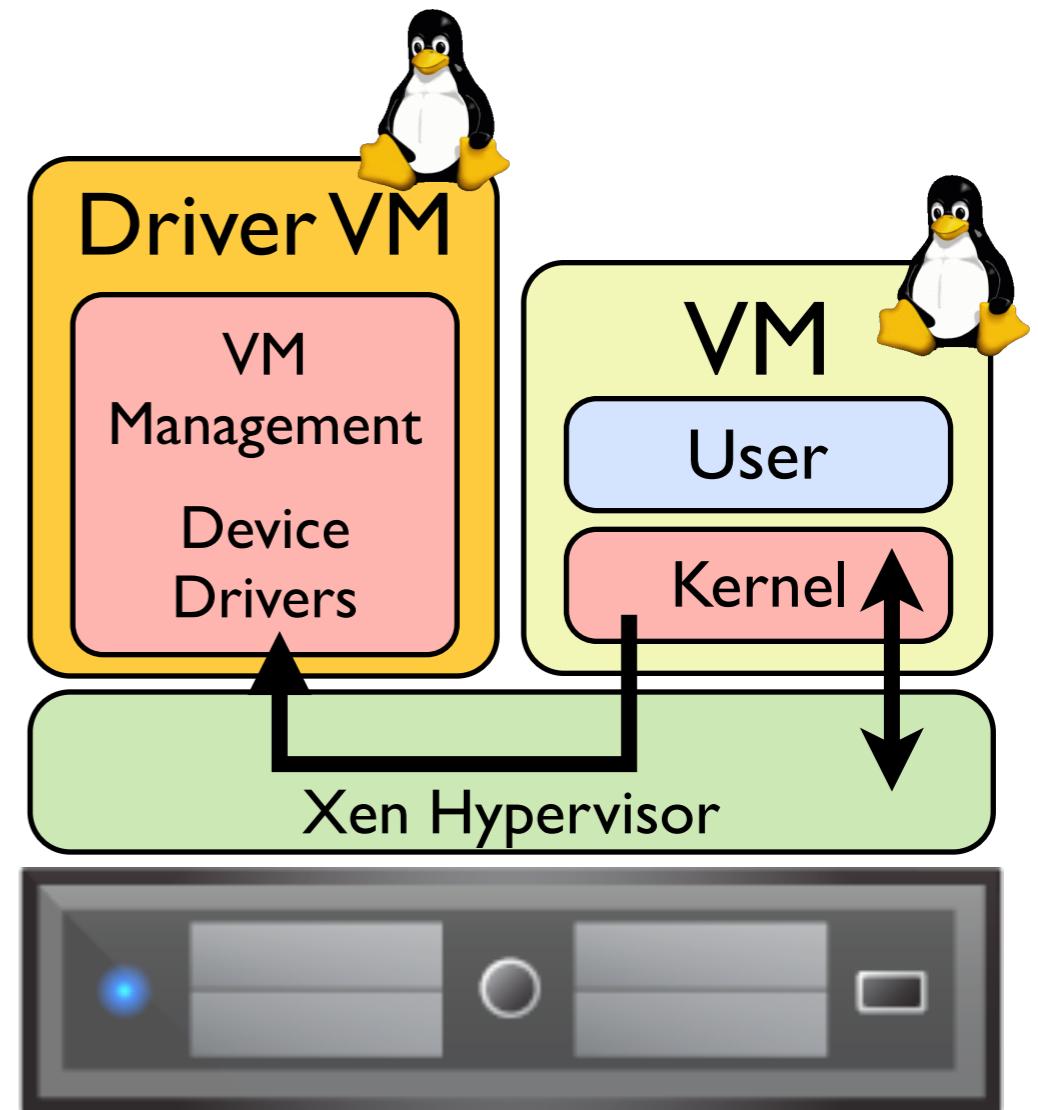
OS asks hypervisor for help to run special instructions

Driver VM is special management VM

- Starts/stops VMs
- Contains Linux device drivers

Very simple hypervisor

- Reduces overhead
- No need for HW virtualization



Trade-offs

Hosted Virtualization

- easier to install, and turn off, great for testing/development
- neg: fewer resources available, because need host OS

Full Virtualization

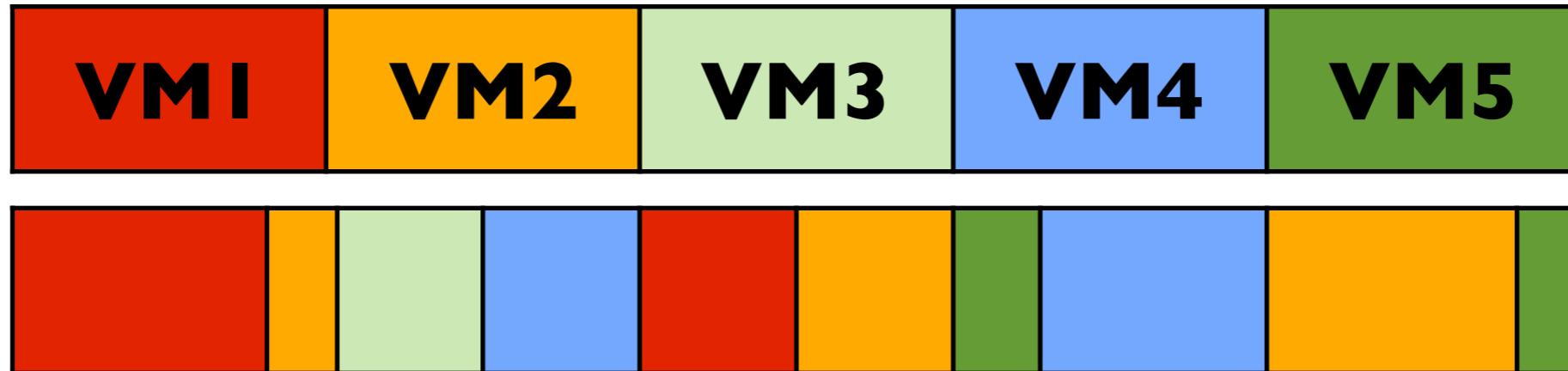
- With or without HW assist
- strong isolation
- greater performance than hosted, better scalability
- neg: needs drivers for all HW

Paravirtualization

- neg: VM is aware it is in a virtual environment (security)
- may be able to optimize b/c it knows it is virtual
- need to modify OS

Virtualizing Memory

System's memory must be shared by all VMs



How should we allocate memory to each VM?

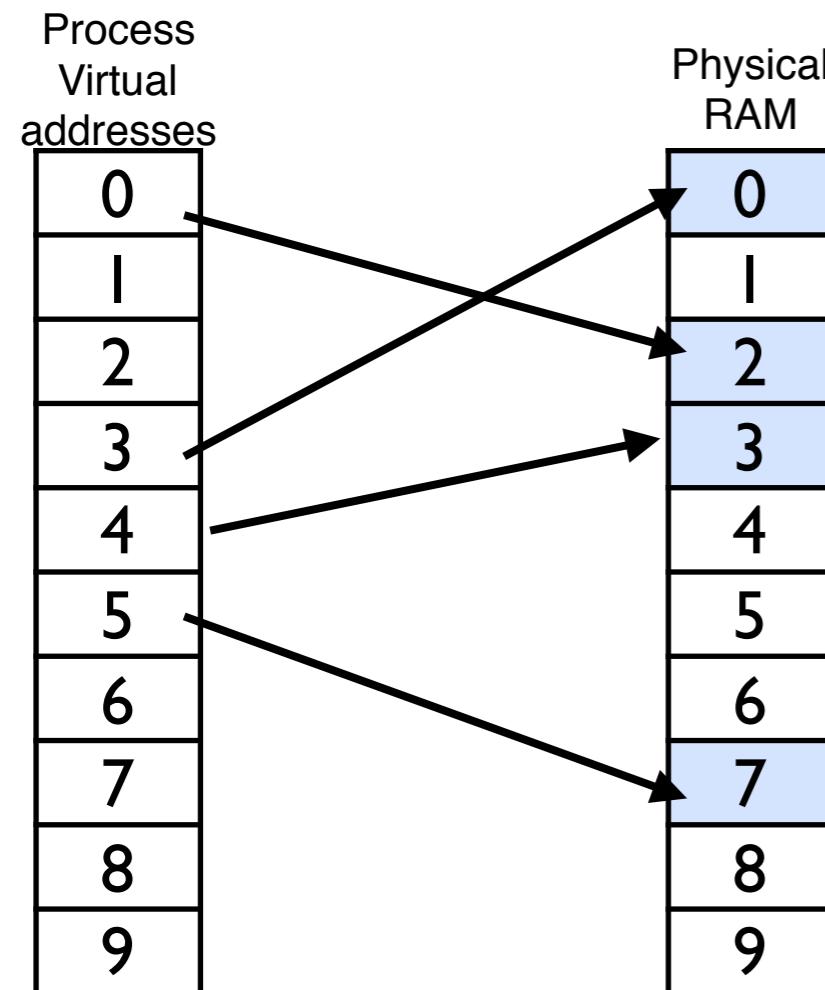
- Contiguous or non-contiguous?

Page tables let us use non-contiguous memory...

- Creating and modifying page tables uses privileged instructions!
- Guest does not even know the real physical addresses!

Page Tables

OS has page table for each process

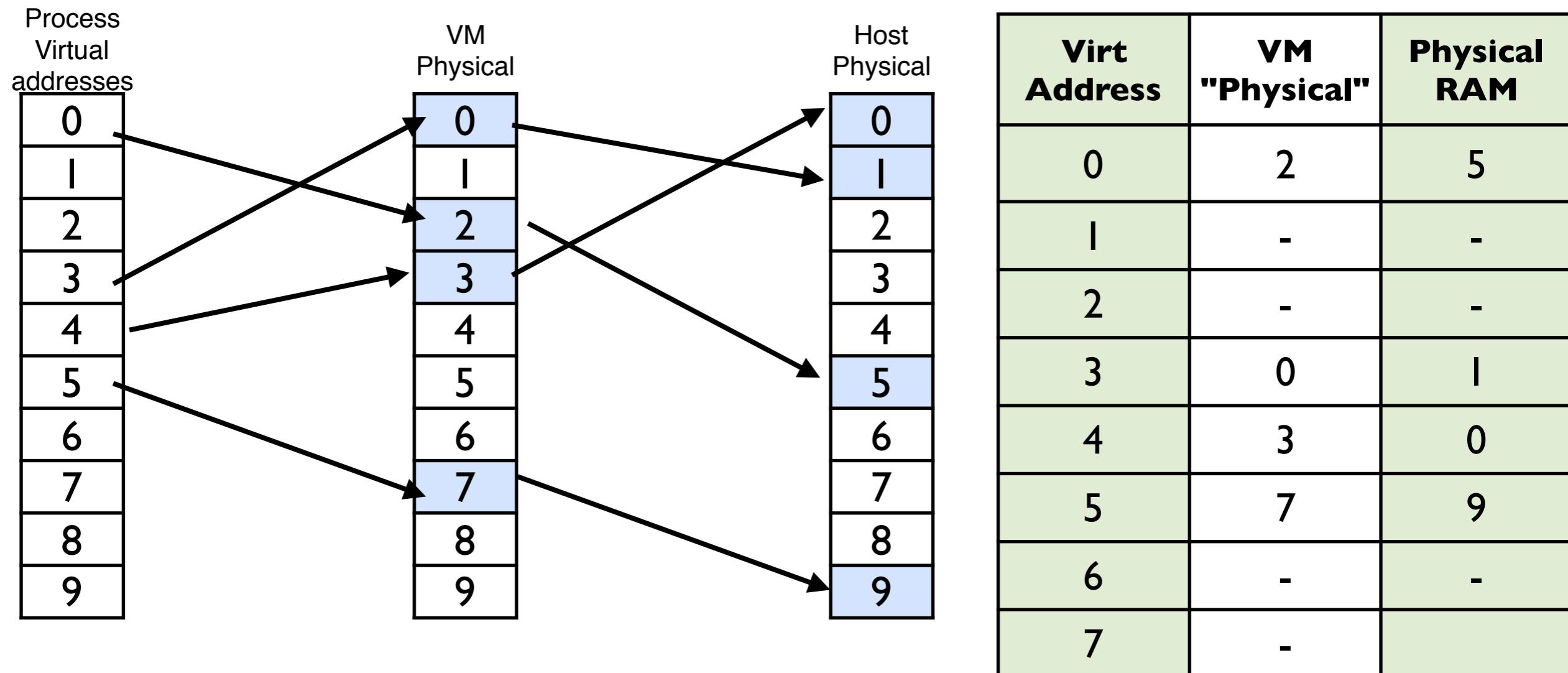


Maps **virtual addresses** to **physical address**

Virt Address	Physical RAM
0	2
1	-
2	-
3	0
4	3
5	7
6	-
7	-

Page Tables

We can do the same thing with VMs

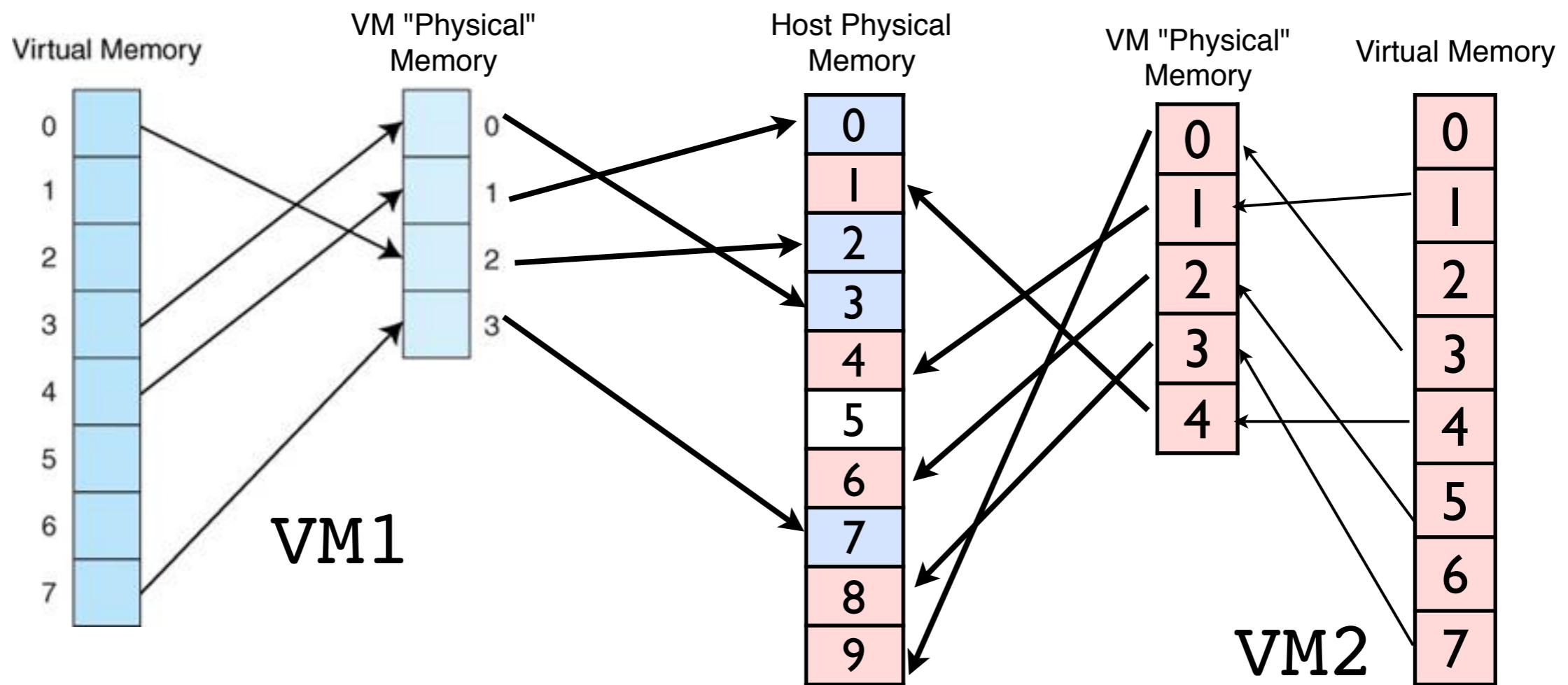


We need another layer of mappings

- Virtual Memory -> VM "Phys" Memory -> Host Phys Memory
- Only the hypervisor knows the true mapping to physical memory

Multiple VMs

Can extend this for multiple VMs



Virtualization layer manages mappings to ensure isolation between VMs and to allocate the right amount of resources to each one

Shadow Page Tables

Shadow Page Tables

- VM's OS thinks it has a regular PT
- Hypervisor adds another translation layer
- Keeps a "shadow" PT with the real mappings

Virt Address	VM "Physical"	Host Physical
0	2	2
1	6	4
2	-	-

Virt Address	VM "Physical"
0	2
1	6
2	-

Virt Address	Host Physical
0	2
1	4
2	-

VM PT

Shadow PT

**MMU / TLB
use this**

Containers

Lightweight virtualization

Process Isolation

Processes

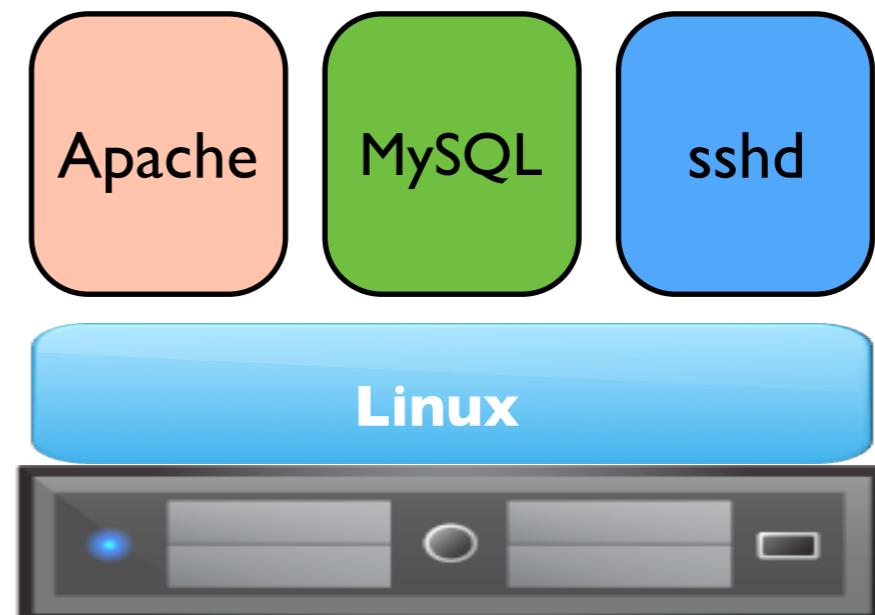
- OS provides isolation

Isolated:

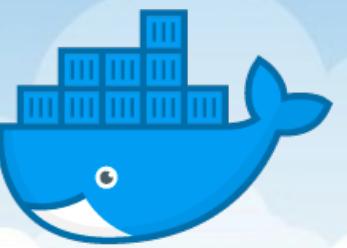
- Memory

Shared:

- File system
- Network
- Devices
- OS Kernel



/etc/
/etc/apache2
/etc/sshd.conf
/etc/mysql
/usr/bin/mysqld
...

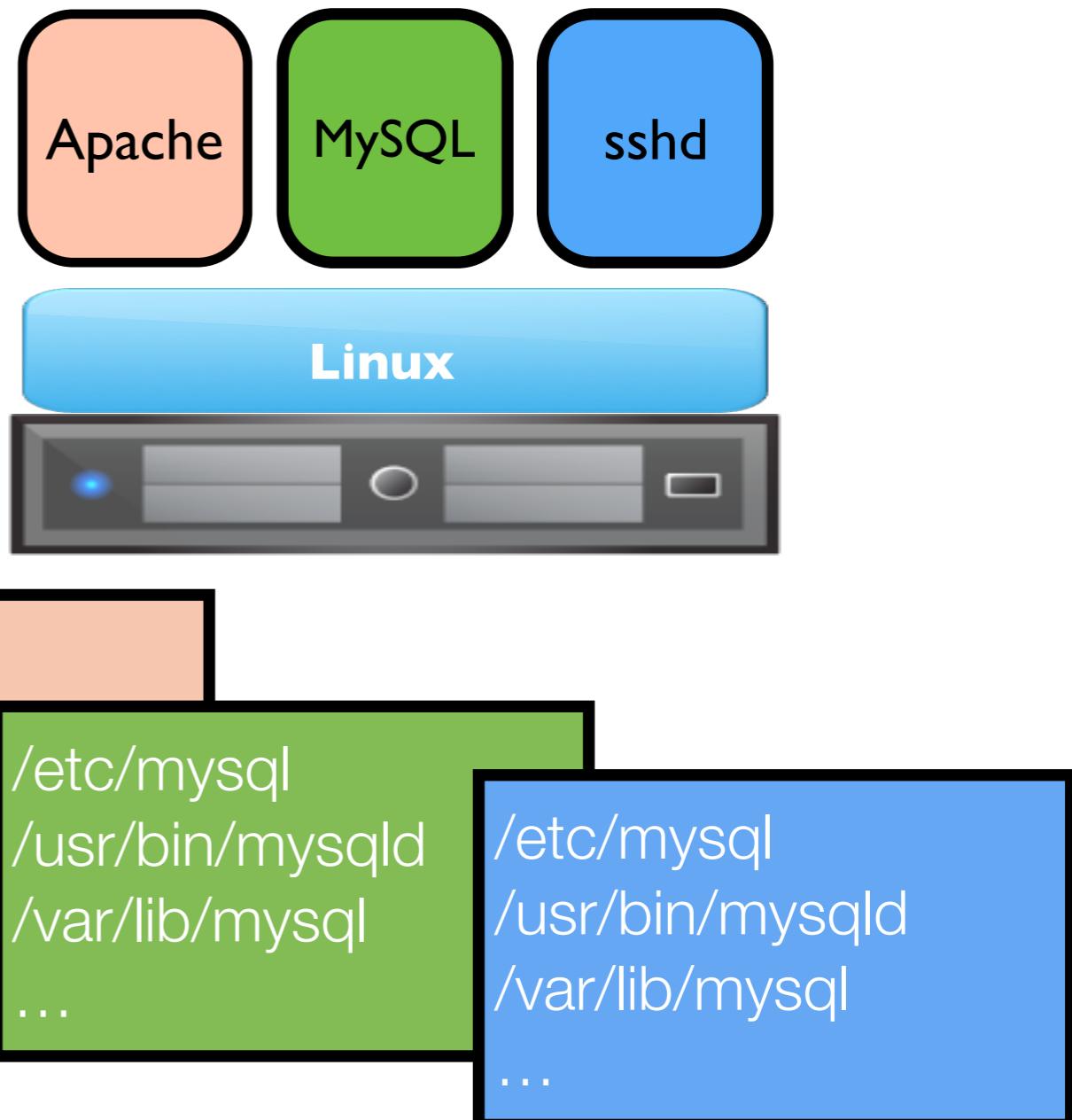


Containers

docker

Containers

- Namespace-based isolation using LXC and cgroups



Isolated:

- Memory
- File system
- Network
- Devices

Shared:

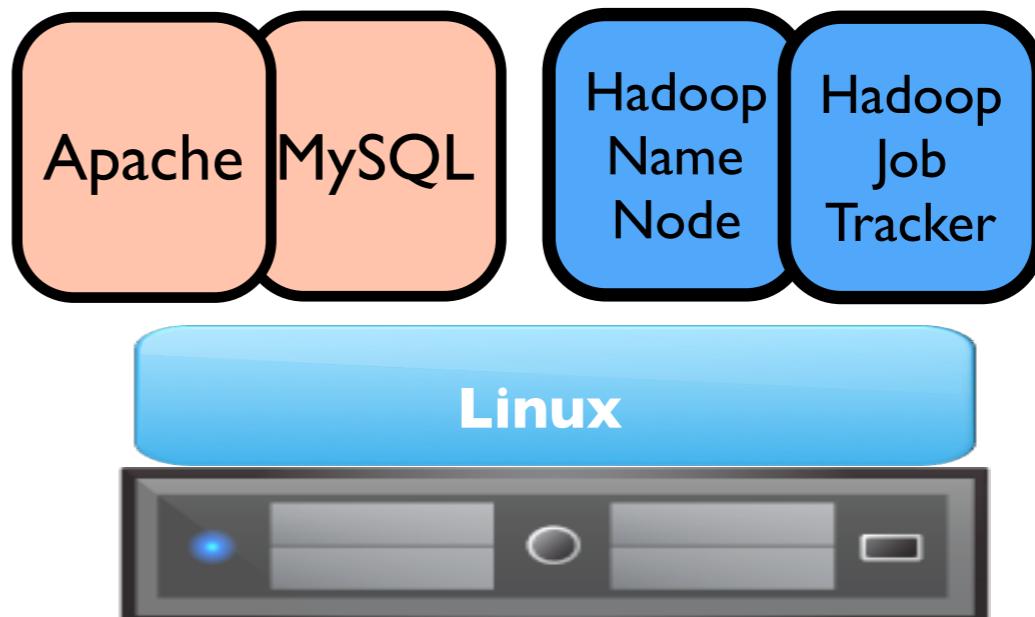
- OS Kernel



Containers

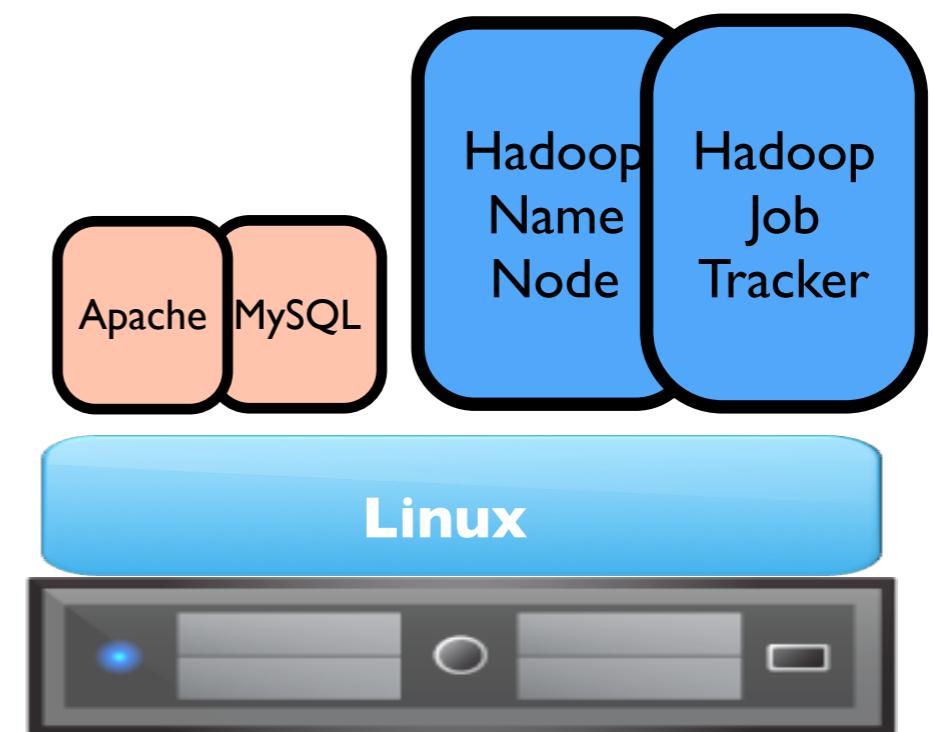
Multi-process containers

- Can run multiple processes in the same container group



Resources:

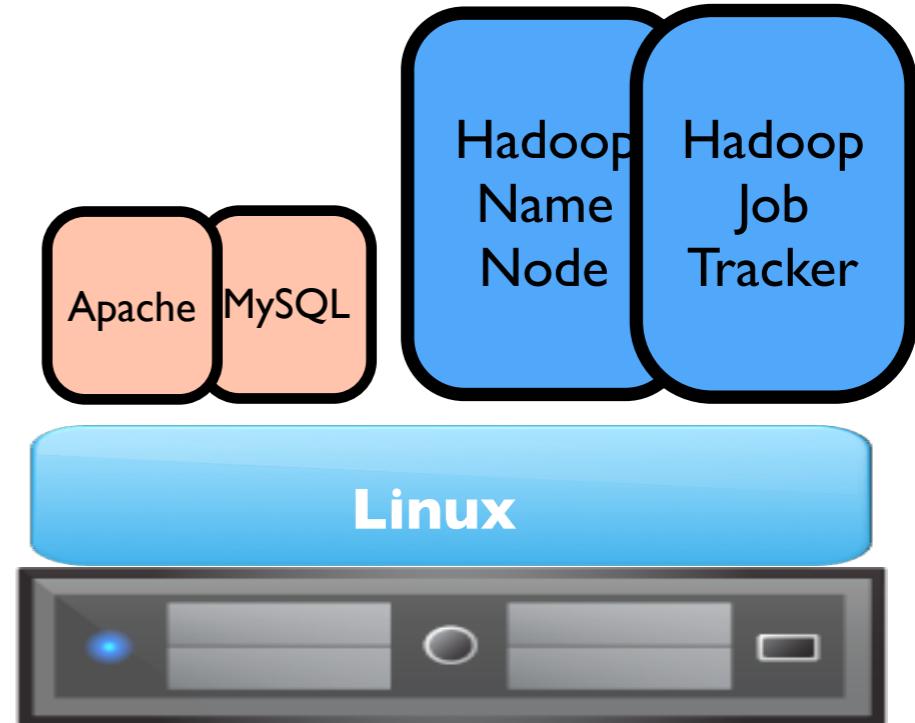
- Can assign CPU weights and memory limits for each group



Shared Kernel

Shared Kernel provides

- Page tables (memory)
- Scheduler (CPU)
- Networking stack
- File system virtualization



What's the difference between the linux kernel and a linux distribution?

- Linux kernel 4.13 vs Linux Kernel 3.5 vs Ubuntu 14.04 vs RedHat 7?

Distro vs Kernel

Kernel = core operating system functionality

- kernel 4.7

Distribution = collection of software and kernel

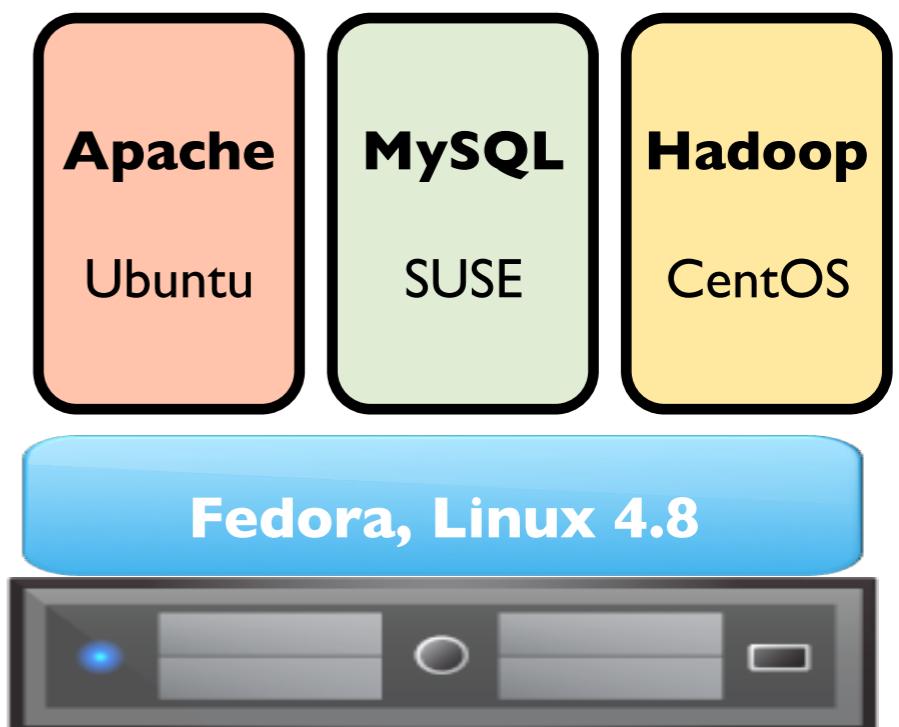
- Ubuntu, CentOS, RedHat

Distributions can work with many different kernels

Containers and Distros

Each container can have its own distribution

Must share the same host kernel



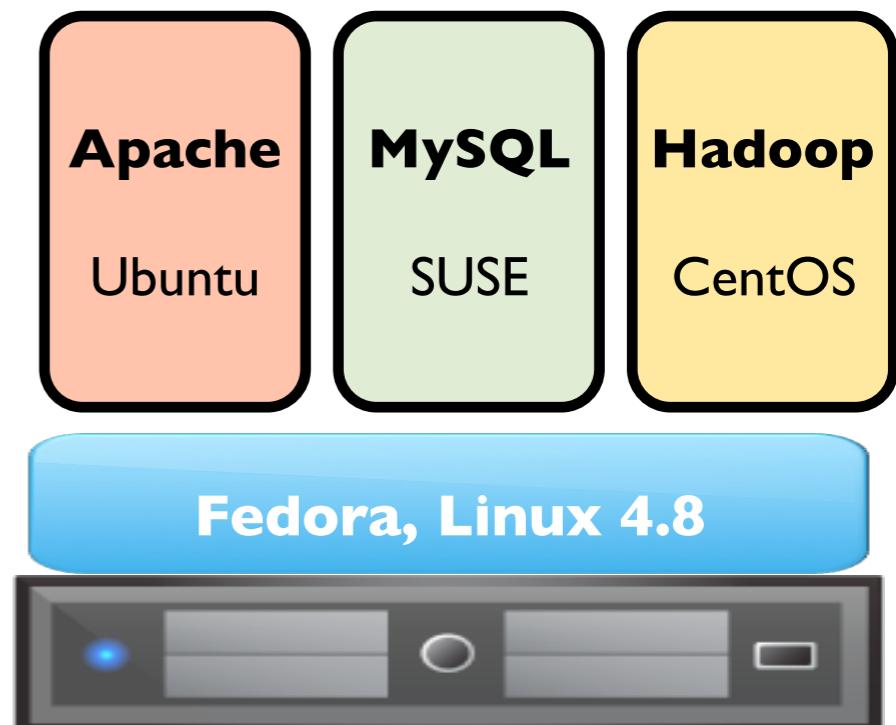
Container Packaging

Deployment - big benefit of containers/virtualization

- Lets you package up an application and all of its requirements
- Even the distribution and 3rd party utilities!
- Very helpful for system administrators

Container “image” includes:

- Linux distribution base files
- Dependency libs/utils
- Configuration files
- Application to run



Does not include...?

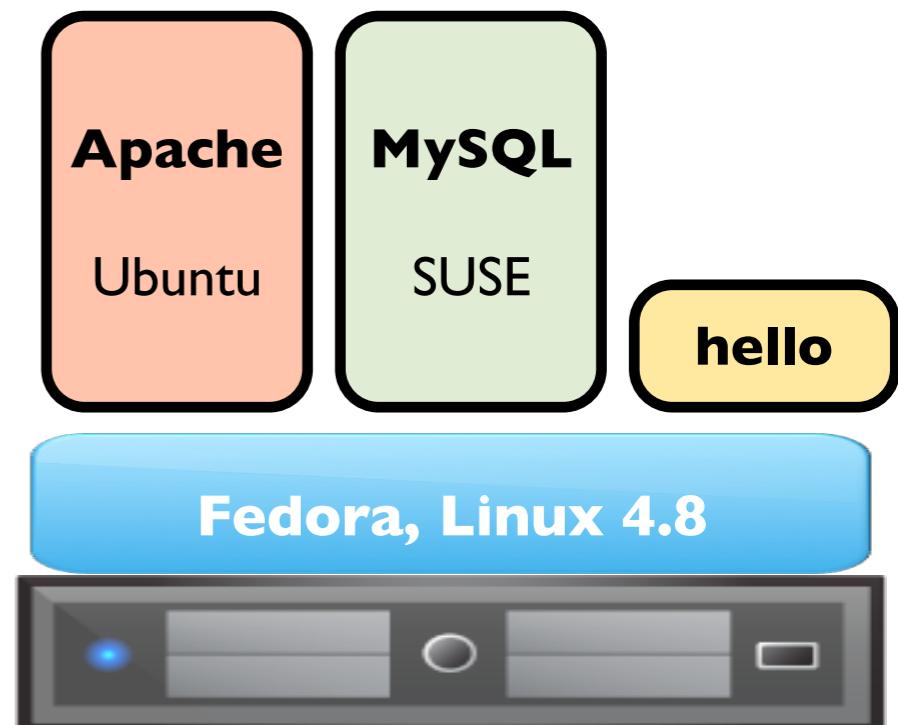
Container Packaging

Deployment - big benefit of containers/virtualization

- Lets you package up an application and all of its requirements
- Even the distribution and 3rd party utilities!
- Very helpful for system administrators

Container “image” includes:

- Linux distribution base files
- Dependency libs/utils
- Configuration files
- Application to run



Can inherit files/libraries from host to reduce size of the container package!

File System Virtualization

Container's file system is built by layering

- Several containers can use the same FS layers

Read/Write

- Allow multiple containers to manipulate data on host FS

Copy on Write

- Each container thinks it has its own version of the FS
- Only duplicate the specific files (data blocks) that are written to

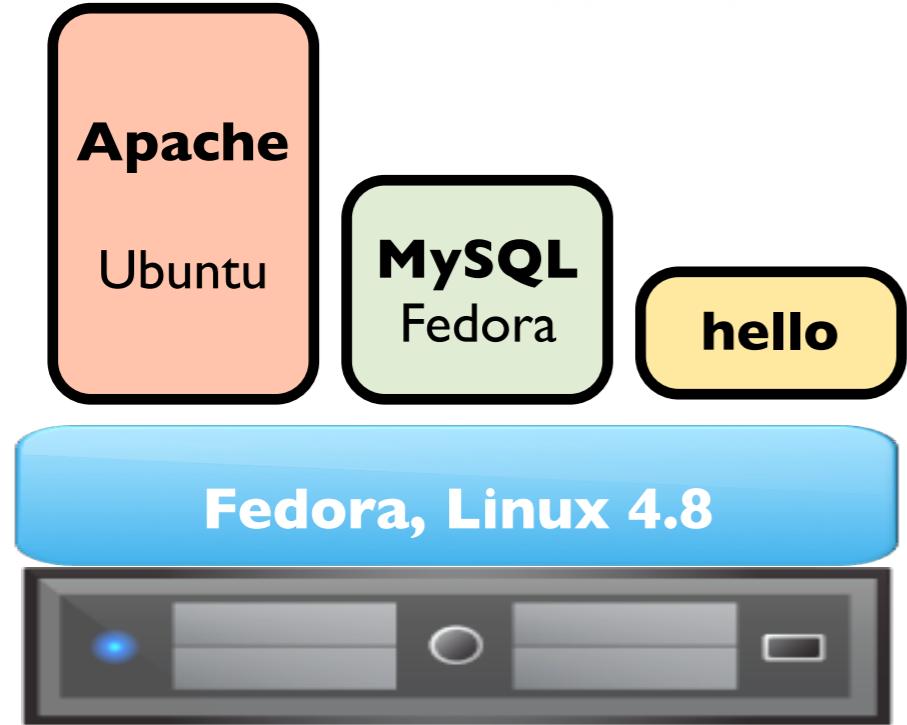
My Hadoop FS

Data Analytics FS

Ubuntu base FS

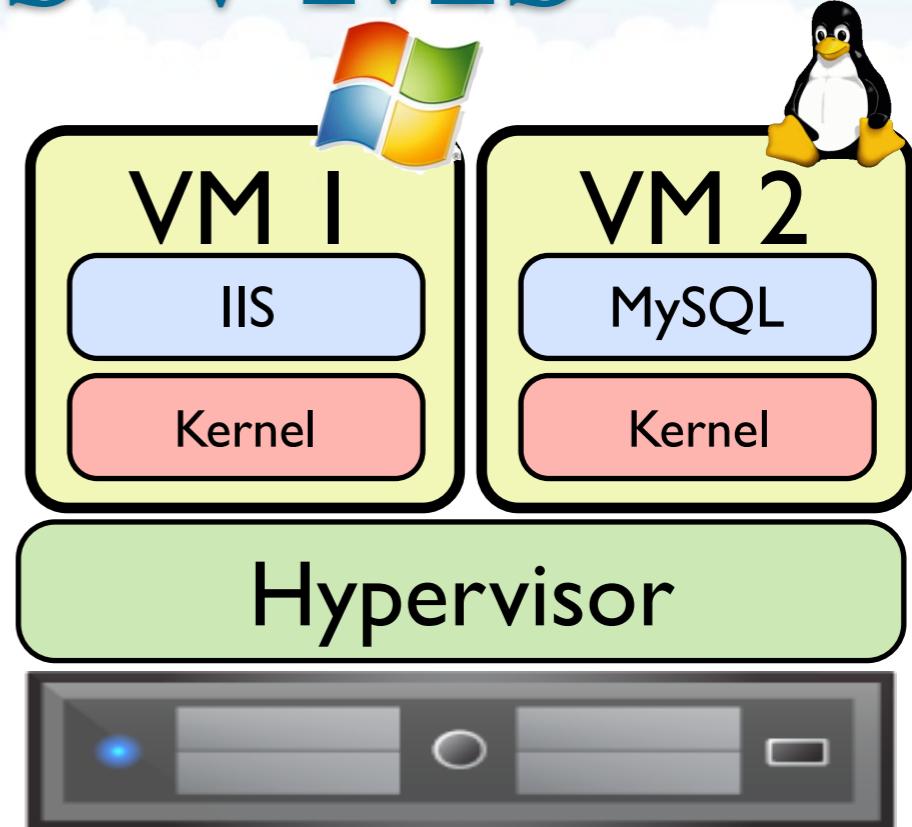
Host FS

Container vs VMs



Pros:

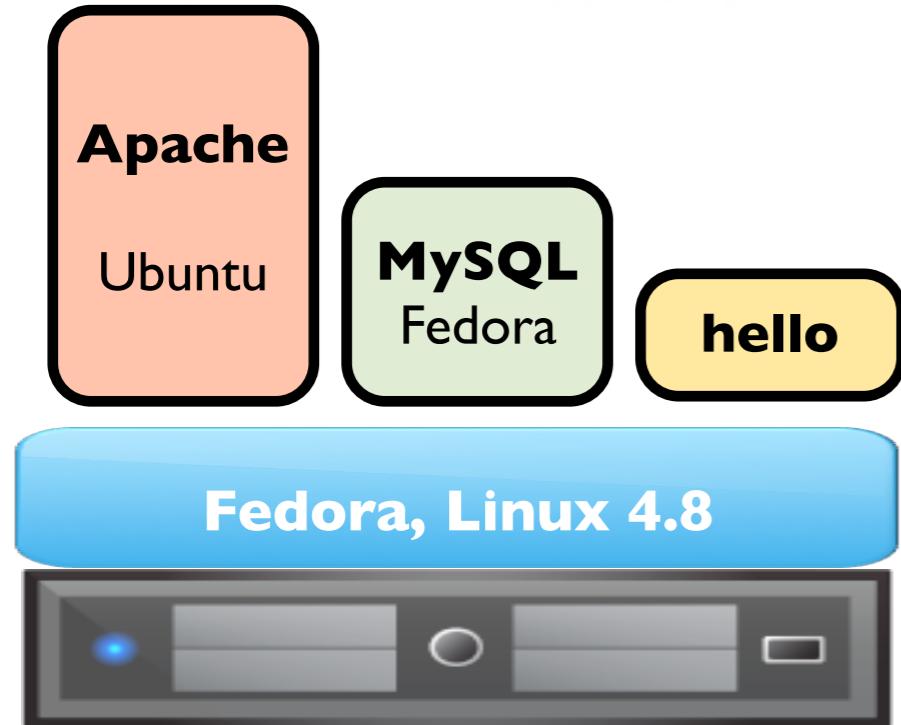
- lightweight (no duplication)
- less resource consumption
- easier to deploy
- specify resources just for application
- startup time



Pros:

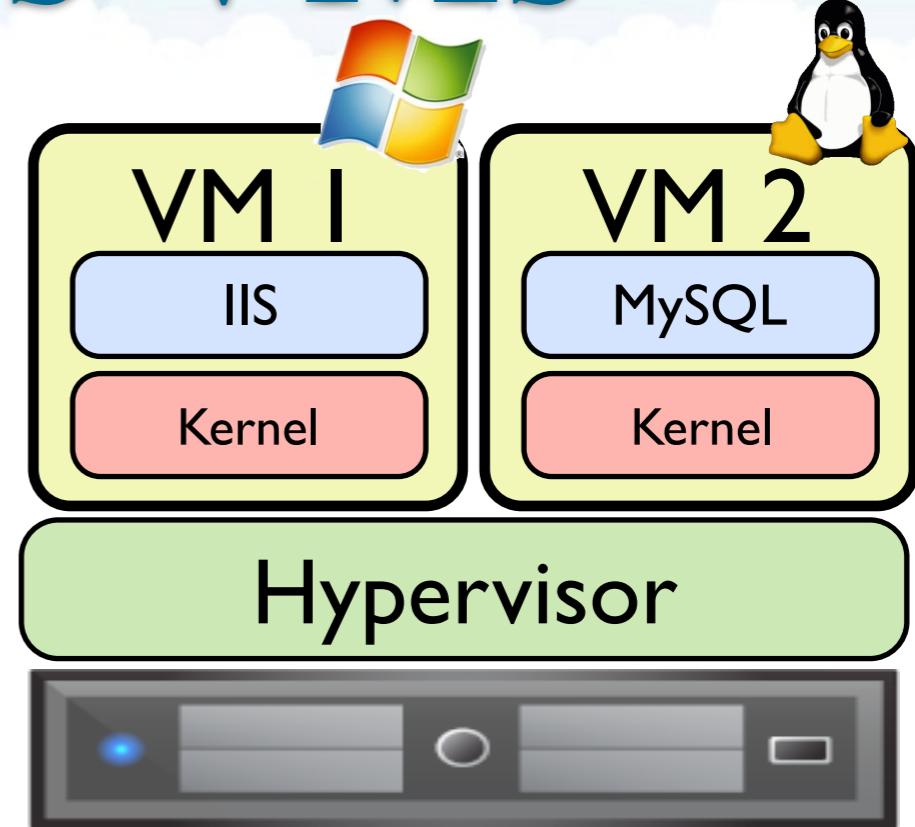
- stronger isolation
- different kernel versions/OSes
- fault tolerance / isolation
- combine with containers

Container vs VMs



Pros:

- ???



Pros:

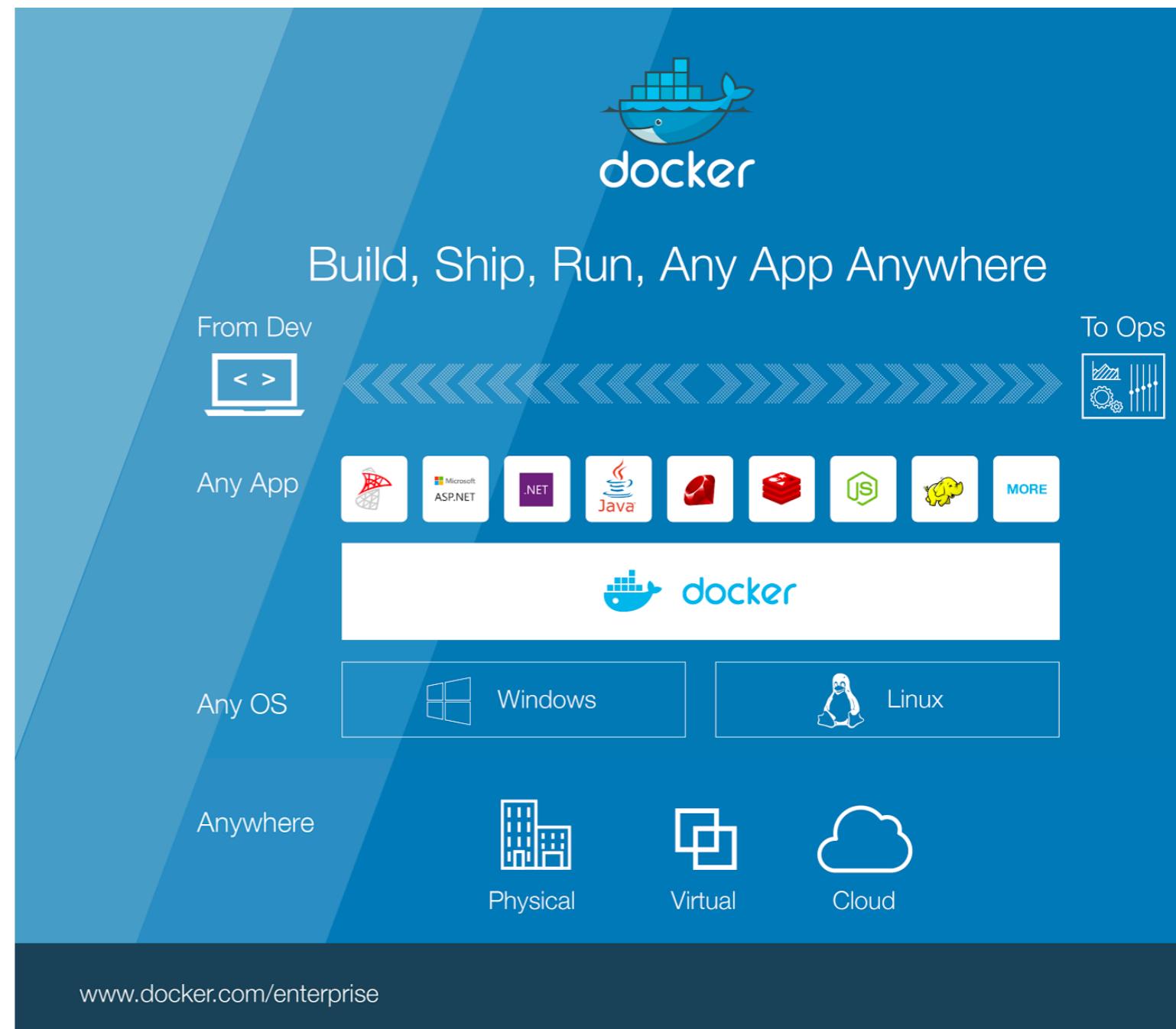
- ???

Containers + VMs

Containers can be combined with virtualization tools

Docker on Windows

- Lets you run windows containers using OS isolation tools
- Lets you run Linux containers by starting a linux VM automatically for you and dividing it up into containers



Distributed Systems Challenges?

Clouds, VMs, Containers

Challenges

Heterogeneity

Openness

Security

Failure Handling

Concurrency

Quality of Service

Scalability

Transparency

Challenges

Heterogeneity: *different HW, SW, workloads*

Openness: *interoperability, shared protocols*

Security: *confidentiality, integrity, availability*

Failure Handling: *crashes, bugs, malicious*

Concurrency: *parallelism, consistency*

Quality of Service: *latency, throughput*

Scalability: *performance gain with more resources*

Transparency: *abstraction layers, interfaces*

Challenges

Heterogeneity

Openness

Security

Failure Handling

Concurrency

Quality of Service

Scalability

Transparency

Clouds

- IaaS
- PaaS
- SaaS

Virtual Machines

Containers

Heterogeneity

Openness

Security

Failure Handling

Concurrency

Quality of Service

Scalability

Transparency

This Course...

1. Lectures

- Focus on fundamental principles and technologies behind distributed systems and the cloud

2. Hands-on Learning

- Focus on practical knowledge of cutting edge tools

This Course...

1. Lectures

- Focus on fundamental principles and technologies behind distributed systems and the cloud

2. Hands-on Learning

- Focus on practical knowledge of cutting edge tools

<https://gwdistsys18.github.io/learn/>

- Docker and Containers
- Big Data and Machine Learning
- Cloud Web Applications

Learn basics of two and the other in depth

- Due October 29th

Sources

<https://www.geekwire.com/2017/amazon-web-services-secret-weapon-custom-made-hardware-network/>

<https://perspectives.mvdirona.com/2010/09/overall-data-center-costs/>

<https://aws.amazon.com/ec2/pricing/on-demand/>

<https://aws.amazon.com/ec2/instance-types/>

<https://www.linkedin.com/pulse/20141118134543-2339144-the-cloud-is-amazon/>

[https://gist.github.com/stevenringo/108922d042c4647f2e195a98e668108a - reInvent 16](https://gist.github.com/stevenringo/108922d042c4647f2e195a98e668108a)

<https://aws.amazon.com/compliance/data-center/data-centers/>

<https://www.zdnet.com/article/aws-cloud-computing-ops-data-centers-1-3-million-servers-creating-efficiency-flywheel/>

<https://www.bloomberg.com/news/2014-11-14/5-numbers-that-illustrate-the-mind-bending-size-of-amazon-s-cloud.html>

[https://youtu.be/AyOAjFNPAbA - reInvent 16 keynote](https://youtu.be/AyOAjFNPAbA)