CS370 Operating Systems

Colorado State University
Yashwant K Malaiya
Fall 2025 L23
HDFS, Virtual Machines

Slides based on
- Text by Silberschatz, Galvin, Gagne
- Various sources

1

CS370 Operating Systems

Colorado State University
Yashwant K Malaiya

Big Data: HDFS and map-reduce

- Various sources, mostly external

Hadoop: Distributed Framework for Big Data

Big Data attributes:

- Large volume: TB -> PB varies with Kryder's law: disk density doubles / 13 months
- Geographically Distributed: minimize data movement
- Needs: reliability, analytic approaches

History:

- Google file system 2003 and Map Reduce 2004 programming lang
- Hadoop to support distribution for the Yahoo search engine project '05, given to Apache Software Foundation '06
- Hadoop ecosystem evolves with Yarn '13 resource management, Pig '10 scripting, Spark '14 distributed computing engine. etc.

- *The Google file system* by Sanjay Ghemawat, Howard Gobioff, and Shun-Tak Leung (2003)
- *MapReduce: Simplified Data Processing on Large Clusters.* by Jeffrey Dean and Sanjay Ghemawat (2004)

Colorado State University

Recent development.

- Big data: multi-terabyte or more data for an app
- Distributed file system
  - Reliability through replication (Fault tolerance)
- Distributed execution
  - Parallel execution for higher performance

Colorado State University

Hadoop: Core components

Hadoop (originally): HDFS + MapReduce

- HDFS: A **d**istributed **f**ile **s**ystem designed to efficiently allocate data across multiple commodity machines, and provide self-healing functions when some of them go down

- MapReduce: A programming framework for processing parallelizable problems across huge datasets using a large number of commodity machines.

  - Commodity machines: lower performance per machine, lower cost, perhaps lower reliability compared with special high-performance machines.

Colorado State University

Challenges in Distributed Big Data

Common Challenges in Distributed Systems

- **Node Failure**: Individual computer nodes may overheat, crash, have hard drive failures, or run out of memory or disk space.

- **Network issues**: Congestion/delays (large data volumes), Communication Failures.

- **Bad data**: Data may be corrupted, or maliciously or improperly transmitted.

- **Other issues**: Multiple versions of client software may use slightly different protocols from one another.

- **Security**

Colorado State University

HDFS Architecture

Hadoop Distributed File System (HDFS):

- HDFS Block size: 64-128 MB   ext4: 4KB

- HDFS file size: "Big"

- Single HDFS FS cluster can span many nodes possibly geographically distributed. datacenters-racks-blades
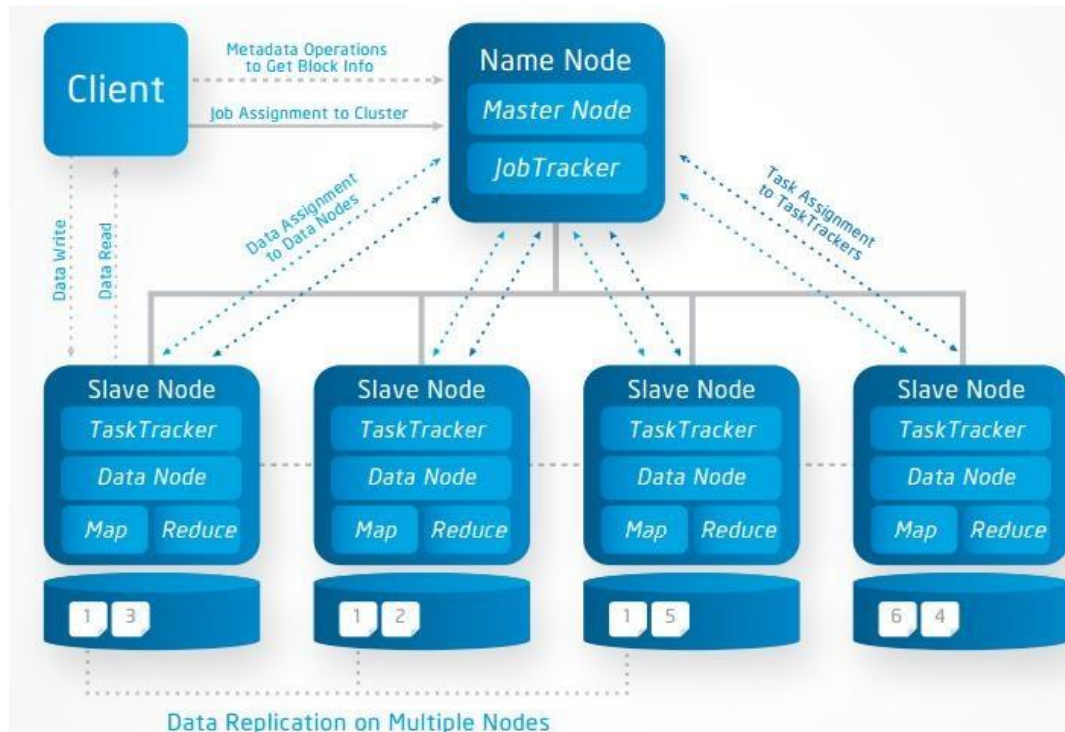
- Node: system with CPU and memory

Metadata (corresponding to superblocks, Inodes)

- Name Node: metadata giving where blocks are physically located

Data (files blocks)

- Data Nodes: hold blocks of files (files are distributed)

Colorado State University
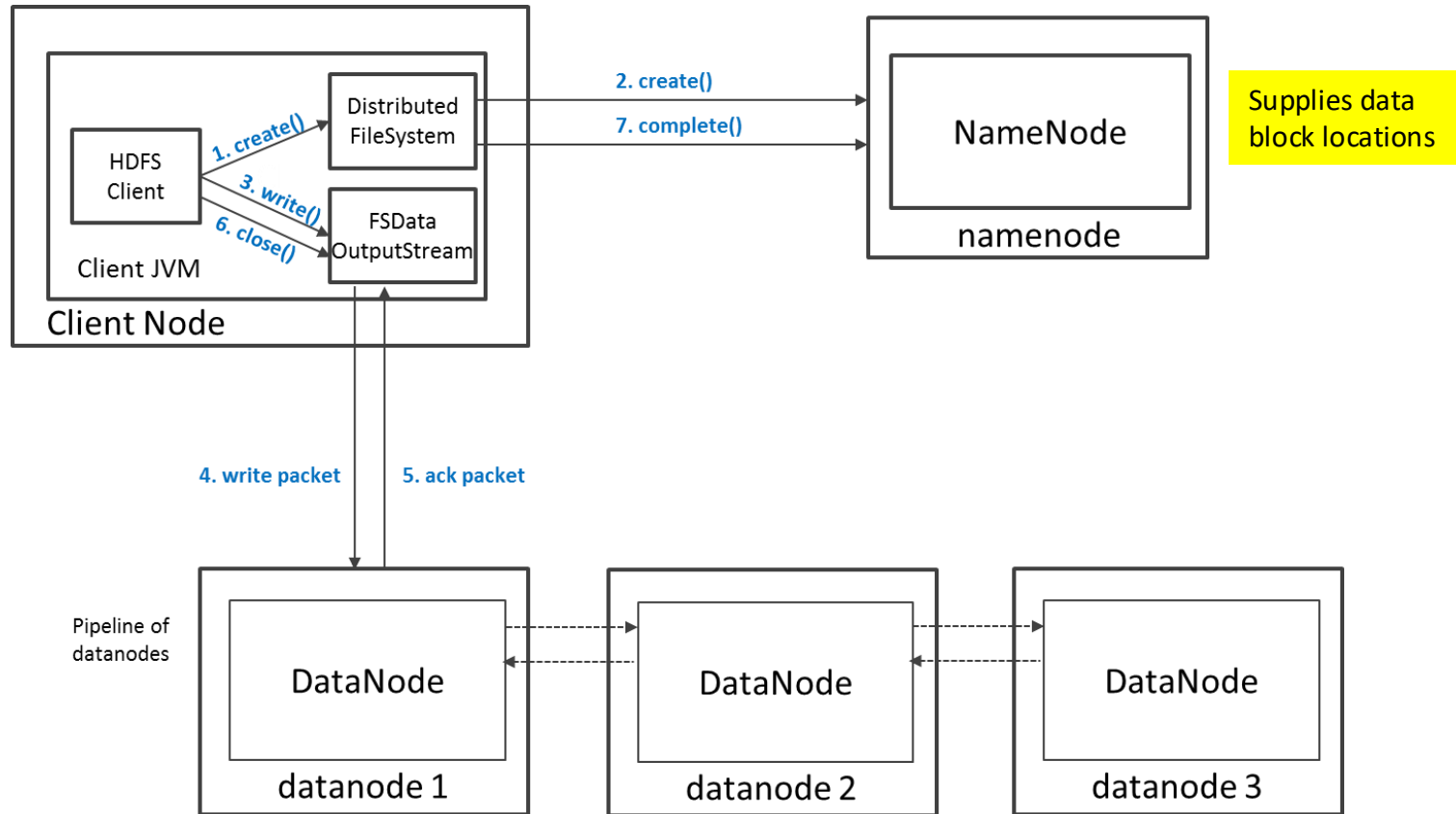
Secondary Name node
If primary fails.

Data is distributed and
replicated.

Name Node: metadata giving
where blocks are physically located
Data Nodes: hold blocks of files
(files are distributed

http://a4academics.com/images/hadoop/Hadoop-Architecture-Read-Write.jpg

Q. What do I need to know? motivation, approaches, concepts

Colorado State University

HDFS Write operation

Client Node

HDFS Client

Client JVM

Distributed FileSystem

1. create()

2. create()

7. complete()

FSData OutputStream

3. write()

6. close()

NameNode

namenode

Supplies data block locations

Pipeline of datanodes

4. write packet

5. ack packet

DataNode

datanode 1

DataNode

datanode 2

DataNode

datanode 3

https://indico.cern.ch/event/404527/contributions/968835/attachments/1123385/1603232/Introduction_to_HDFS.pdf

CERN

Colorado State University

HDFS Fault-tolerance

- Disks use *error detecting codes* to detect corruption.

- Individual node/rack may fail.

- Data Nodes (on slave nodes):

  - data is replicated. Default is 3 times. Keep a copy far away.

  - Send periodic heartbeat (I'm OK) to Name Nodes. Perhaps once every 10 minutes.

  - Name node creates another copy if no heartbeat.

Colorado State University

Name Node (on master node) Protection:

- Transaction log for file deletes/adds, etc. Creation of more replica blocks, when necessary, after a Data Node failure

- Standby name node: namespace backup
  - In the event of a failover, the Standby will ensure that it has read all of the edits from the Journal Nodes and then promotes itself to the Active state

  - Implementation/delay version dependent

Name Node metadata is in RAM as well as checkpointed on disk.
On disk the state is stored in two files:
- fsimage: Snapshot of file system metadata
- editlog: Changes since last snapshot

Colorado State University

HDFS Command line interface

- hadoop fs –help

- hadoop fs –ls : List a directory

- hadoop fs mkdir : makes a directory in HDFS

- hadoop fs –rm : Deletes a file in HDFS

- copyFromLocal : Copies data to HDFS from local filesystem

- copyToLocal : Copies data to local filesystem

- Java code can read or write HDFS files (URI) directly

HDFS is on top of a local file system

https://hadoop.apache.org/docs/r2.4.1/hadoop-project-dist/hadoop-common/FileSystemShell.html

Colorado State University

MapReduce Engine:

- JobTracker splits up the job into smaller tasks("Map") and sends it to the TaskTracker process in each node.

- TaskTracker reports back to the JobTracker node and reports on job progress, sends partial results ("Reduce") or requests new jobs.

- Tasks are run on local data, thus avoiding movement of bulk data.

- Originally developed for search engine implementation.

Colorado State University

Fault-tolerance in Computers

- Computers are inherently prone to faults
  - Soft errors are very common
  - Bit flipping its binary value
  - Corruption due to noise
  - Corruption in transmission or recording
- Large amounts of data: requires various fault-tolerance approaches
- Music players, Internet, Banking, Flight control, etc.

Colorado State University

- Hadoop YARN: A framework for job scheduling and cluster resource management, can run on top of Windows Azure or Amazon S3.
- **Apache spark is more general, faster and easier to program than MapReduce**.

Resilient Distributed Datasets: A Fault-Tolerant Abstraction for In-Memory Cluster Computing, Berkeley, 2012

CS370 Operating Systems

Colorado State University
Yashwant K Malaiya

Virtual Machines

Slides based on
- Text by Silberschatz, Galvin, Gagne
- Various sources

Virtualization

- Why we need virtualization?
- The concepts and terms
- Brief history of virtualization
- Types of virtualization
- Implementation Issues
- Containers

Ch 18 + external

"I ordered a computer from this company and this is all they sent me."

We will skip implementation specific details. Please consult the documentation and watch related videos.

Colorado State University

Isolation and resource allocation

Isolation levels:

- **Process**: Isolated address space
- **Container**: Isolated set of processes, files and network
- **Virtual Machines** (VM): Isolated OSs
- **Physically isolated** machines

Resource allocation:

- Resources need to be allocated to
  - processes
  - Containers
  - VMs and
- managed to serve needs best.

Colorado State University

Virtualization in Virtual machines

- A Virtual scheme provides a simpler perspective of a Physical scheme. Needs mapping.
  - Example: each process a separate virtual address space.
  - OS allocates physical memory and disk space and handles mapping.
- System ("machine") virtualization
  - A machine needs its own CPU, memory, storage, I/O to run its OS and apps. "Machine" = {CPU, memory, storage, I/O, OS, apps}
  - Needs to be isolated from other machines.
  - "Virtual machines" allocated part of resources from physical machine (hardware) with allocation done by a Virtual Machine Monitor (VMM) or hypervisor.
  - A single physical machine can run multiple virtual machines.
  - A virtual machine can be "migrated" from one physical system to another.

Colorado State University

"Tell that intern that you can't migrate physical machines."
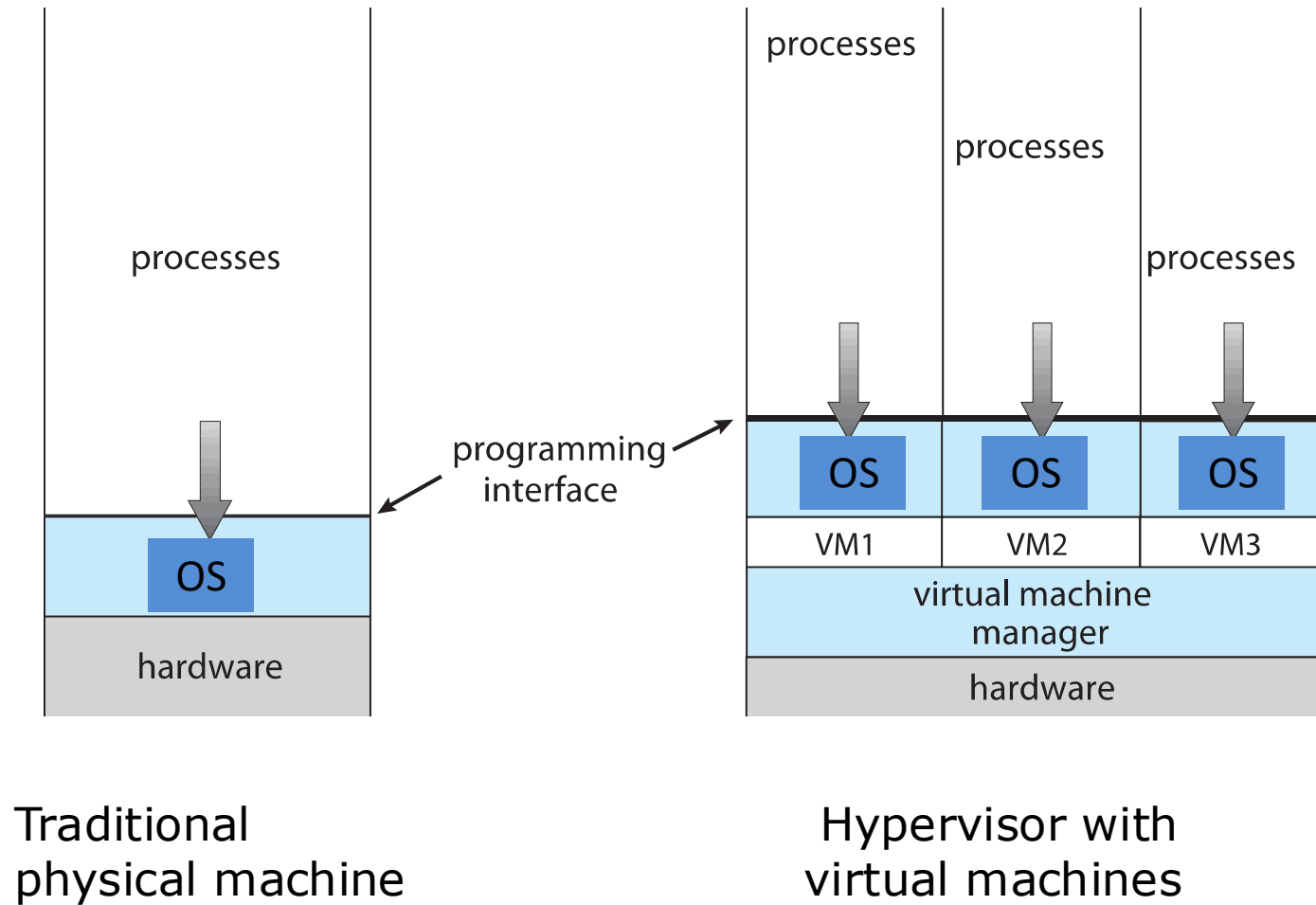
Colorado State University

Virtualization

- Processors have gradually become very powerful

- Dedicated servers can be very underutilized (5-15%)

- Virtualization allow a single server to support several virtualized servers: typical consolidation ratio 6:1

- Power cost a major expense for data centers
  - Companies frequently locate their data centers in the middle of nowhere where power cost is low

- If a hardware server crashes, would be nice to migrate the load to another one.

- A key component of cloud computing

Colorado State University

Virtual Machines (VM)

- **Virtualization** technology enables a single PC/server to simultaneously run multiple Virtual Machines,
  - with different operating systems or multiple sessions of a single OS.

- A machine with virtualization can host many applications, including those that run on different operating systems, on a single platform.

- The host operating system can support a number of virtual machines, each of which has the characteristics of a specific OS.

- The software that enables virtualization is a **virtual machine monitor (VMM),** or **hypervisor.**

Colorado State University

Traditional
physical machine

Hypervisor with
virtual machines

Colorado State University

Kinds of Virtual Systems

Virtualization

- Hypervisor based
  - Full virtualization: bare metal hypervisor
  - Para virtualization: modified guest OS
  - Host OS virtualization
- Container system: multiple user space instances
- Environment virtualization
  - Java virtual machine, Dalvic virtual machine
- Software simulation of hardware/ISA
  - Android JDK
  - SoftPC etc.
- Emulation using microcode

Colorado State University

Brief history

- Early 1960s IBM experimented with two independently developed hypervisors - SIMMON and CP-40
- Common CPU **modes**: **user** and **supervisor** (*Privileged)*
- In 1974, Popek and Goldberg published a paper which listed what conditions a computer architecture should satisfy to support virtualization efficiently
  - *Privileged instructions: Those that trap if the processor is in user mode and do not trap if it is in system mode (supervisor mode).*
  - *Sensitive instructions: that attempt to change the configuration of resources in the system or whose behavior or result depends on the configuration of resources*
  - Theorem. For any conventional third-generation computer, an effective VMM may be constructed if the set of sensitive instructions for that computer is a subset of the set of privileged instructions.
  - The x86 architecture that originated in the 1970s did not meet these for requirements for decades.
-

Colorado State University

"Strictly Virtualizable"

A processor or mode of a processor is *strictly virtualizable* if, when executed in a lesser privileged mode:

- all instructions that access privileged state trap

- all instructions either trap or execute identically

- Satisfied by modern CPUs (with virtualization option enabled).
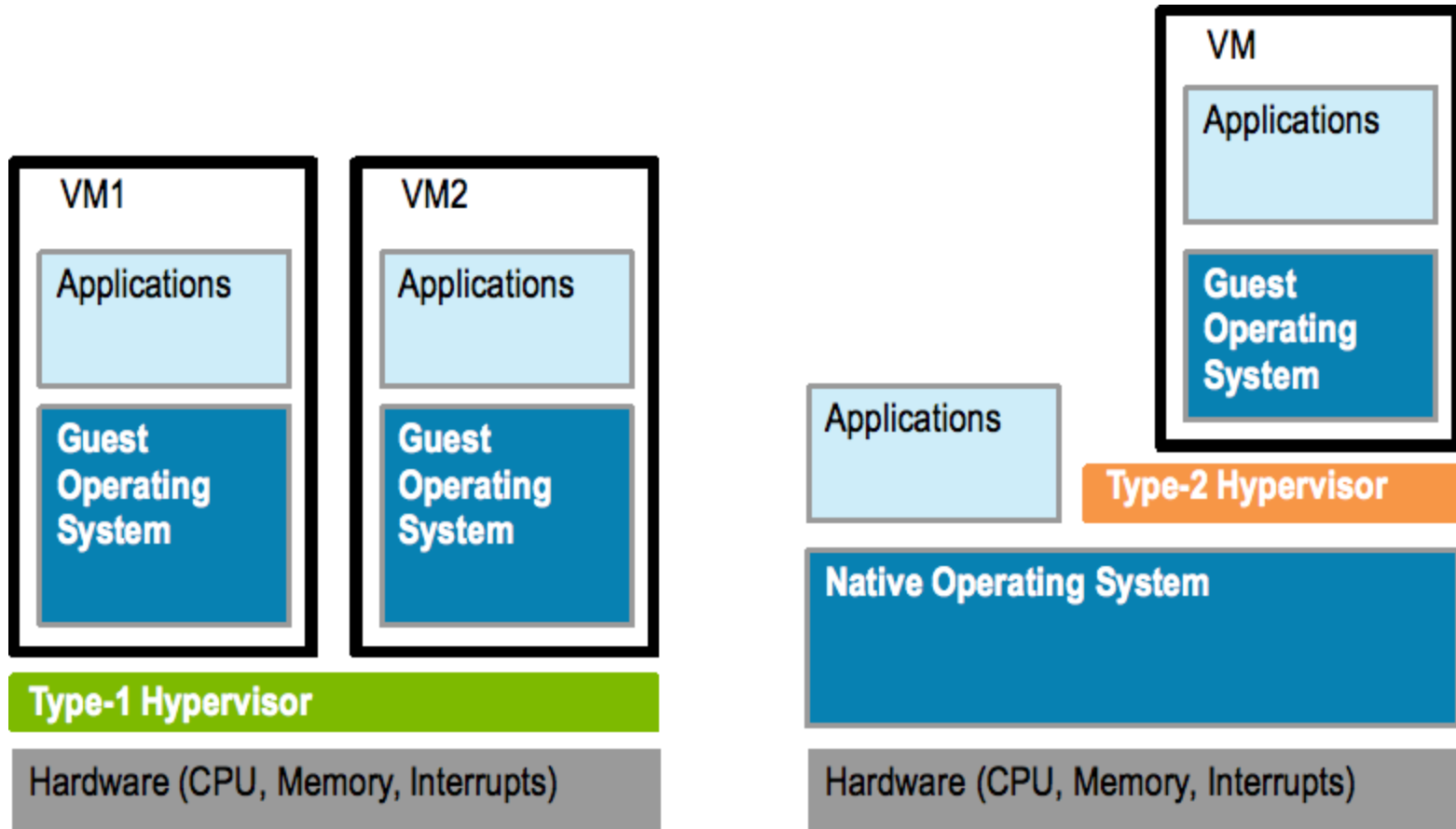
Colorado State University

Brief history (recent)

- Stanford researchers developed a new hypervisor and then founded VMware
  - first virtualization solution for x86 in 1999
  - Linux, windows
- Others followed
  - Xen, 2003   University of Cambridge, Xen Project community
  - KVM, 2007 startup/Red Hat
  - VirtualBox (Innotek GmbH/Sun/Oracle) , 2007
  - Hyper-V (Microsoft), 2008
- Cgroups (2007 Google), Docker Engine 2013

Colorado State University

Implementation of VMMs

- **Type 1 hypervisors** - Operating-system-like software built to provide virtualization. Runs on 'bare metal".
  - Including VMware ESX, Joyent SmartOS, and Citrix XenServer
- Also includes general-purpose operating systems that provide standard functions as well as VMM functions
  - Including Microsoft Windows Server with HyperV and RedHat Linux with KVM
- **Type 2 hypervisors** - Applications that run on standard operating systems but provide VMM features to guest operating systems
  - Including VMware Workstation and Fusion, Parallels Desktop, and Oracle VirtualBox
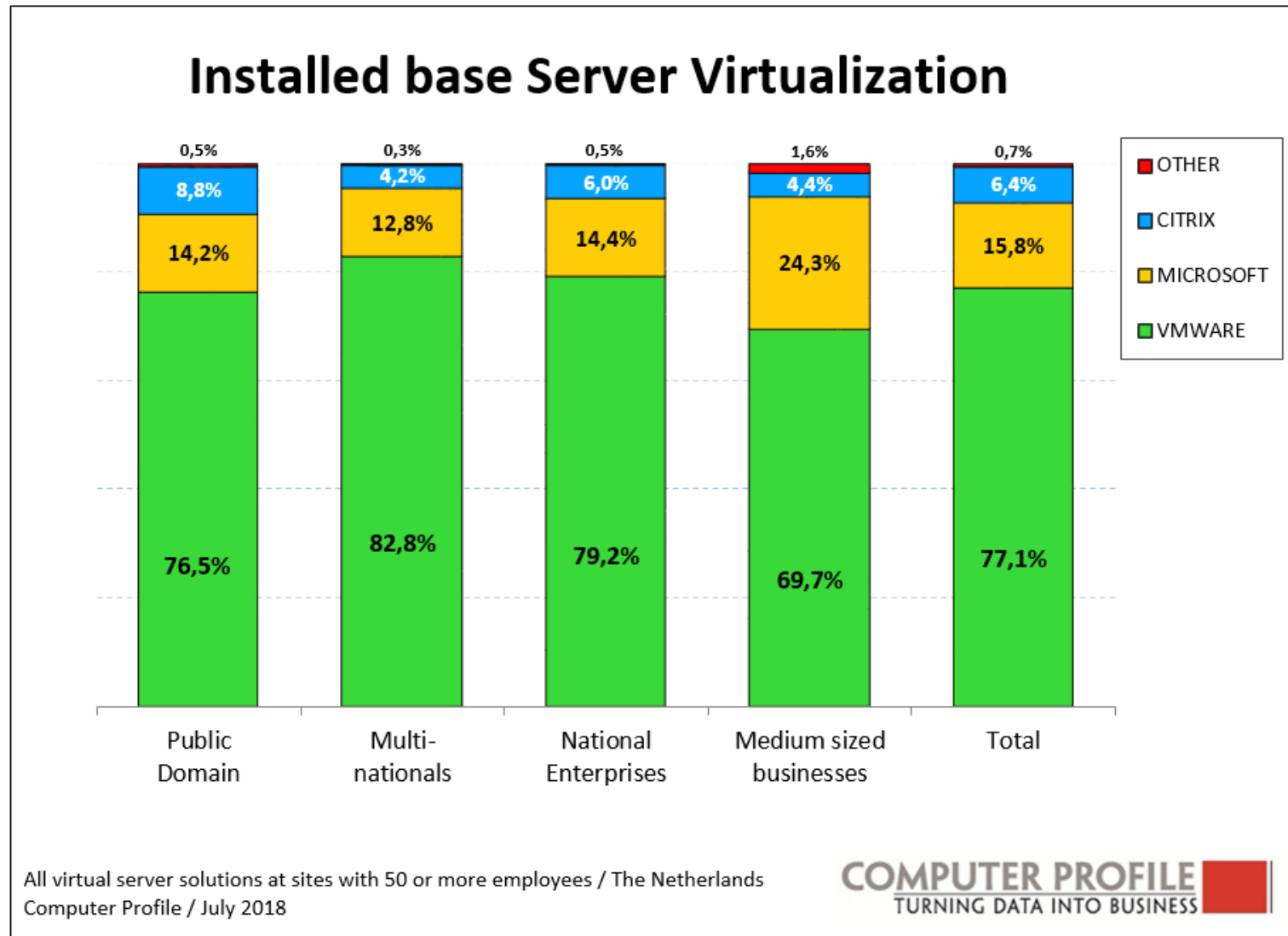
Colorado State University

A higher layer uses services of the lower layers.

https://microkerneldude.files.wordpress.com/2012/01/type1-vs-2.png

Colorado State University

Installed base Server Virtualization

Legend:
- OTHER
- CITRIX
- MICROSOFT
- VMWARE

Category	Public Domain	Multi-nationals	National Enterprises	Medium sized businesses	Total
OTHER	0,5%	0,3%	0,5%	1,6%	0,7%
CITRIX	8,8%	4,2%	6,0%	4,4%	6,4%
MICROSOFT	14,2%	12,8%	14,4%	24,3%	15,8%
VMWARE	76,5%	82,8%	79,2%	69,7%	77,1%

All virtual server solutions at sites with 50 or more employees / The Netherlands
Computer Profile / July 2018

COMPUTER PROFILE
TURNING DATA INTO BUSINESS

Colorado State University
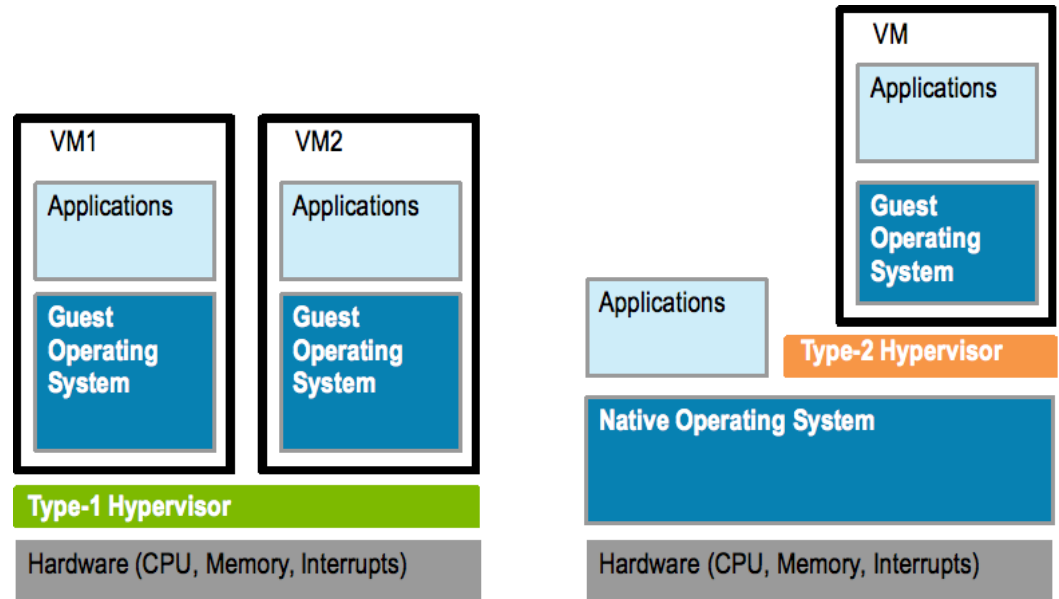
30

User mode and Kernel (supervisor) mode

- Special instructions:
- Depending on whether it is executed in kernel/user mode
  - "Sensitive instructions"
- Some instructions cause a trap when executed in user-mode
  - "Privileged instructions"
- A machine is virtualizable only if sensitive instructions are a subset of privileged instructions
  - Intel's 386 did not always do that. Several sensitive 386 instructions were ignored if executed in user mode.

- Fixed in 2005 virtualization may need to be enabled using BIOS

  - Intel CPUs: VT (Virtualization Technology)
  - AMD CPUs: SVM (Secure Virtual Machine)

Colorado State University

Virtualization support

- Terminology:
  - Guest Operating System
    - The OS running on top of the hypervisor
  - Host Operating System
    - For a type 2 hypervisor: the OS that runs on the hardware ¨executions

- Create environments in which VMs can be run

- When a guest OS is started in an environment, continues to run until it causes an exception and traps to the hypervisor
  - For e.g., by executing an I/O instruction

- Set of operations that trap is controlled by a hardware bit map set by hypervisor
  - trap-and-emulate approach becomes possible

Colorado State University

Implementation of VMMs

What question do you see here?

- What mode does hypervisor run in? Guest OSs?

- Are Guest OSs aware of hypervisor?

- How is memory managed?

- How do we know what is the best choice? Answers coming up.

Colorado State University

Virtual Machine (VM) as a software construct

- Each VM is configured with some number of processors, some amount of RAM, storage resources, and connectivity through the network ports.

- Once the VM is created it can be activated on like a physical server, loaded with an operating system and software solutions, and used just like a physical server.

- Unlike a physical server, VM only sees the resources it has been configured with, not all of the resources of the physical host itself.

- The hypervisor facilitates the translation and I/O between the virtual machine and the physical server.

Colorado State University

Virtual Machine (VM) as a set of files

- Configuration file describes the attributes of the virtual machine containing
  - server definition,
  - how many virtual processors (vCPUs)
  - how much RAM is allocated,
  - which I/O devices the VM has access to,
  - how many network interface cards (NICs) are in the virtual server
  - the storage that the VM can access
- When a virtual machine is instantiated, additional files are created for logging, for memory paging etc.
- Copying a VM produces not only a backup of the data but also a copy of the entire server, including the operating system, applications, and the hardware configuration itself

Colorado State University

Virtualization benefits

- Run multiple OSes on a single machine
  - **Consolidation**, app dev, …
- Security: Host system protected from VMs; VMs protected from each other
  - Sharing though shared file system volume, network communication
- Freeze, suspend, running VM
  - Then can move or copy somewhere else and **resume**
    - **Live migration**
  - Snapshot of a given state, able to restore back to that state
  - **Clone** by creating copy and running both original and copy
- Hence – cloud computing

Colorado State University

- VM needs two modes: both in real user mode
  - virtual user mode and virtual kernel mode
- When Guest OS attempts to execute a privileged instruction, what happens?
  - Causes a trap
  - VMM gains control, analyzes error, executes operation as attempted by guest
  - Returns control to guest in user mode
  - Known as **trap-and-emulate**
- Trap-and-emulate was the technique used for implementing floating point instructions in CPUs without floating point coprocessor

Colorado State University

Handling sensitive instructions

- Some CPUs didn't have clean separation between privileged and non-privileged instructions
  - Sensitive instructions
    - Consider Intel x86 `popf` instruction
    - If CPU in privileged mode -> all flags replaced
    - If CPU in user mode -> on some flags replaced
      - No trap is generated
- Binary translation (complex) solves the problem
  1. If guest VCPU is in user mode, guest can run instructions natively
  2. If guest VCPU in kernel mode (guest believes it is in kernel mode)
     1. VMM examines every instruction guest is about to execute by reading a few instructions ahead of program counter
     2. Special instructions translated into new set of instructions that perform equivalent task (for example changing the flags in the VCPU)
  3. Cached translations can reduce overhead
- Not needed in newer processors with virtualization support.

Colorado State University

Type 1 Hypervisors

- Run on top of *bare metal*
- Guest OSs believe they are running on bare metal, are unaware of hypervisor
  - are not modified
  - Better performance
- Choice for data centers
  - Consolidation of multiple OSes and apps onto less HW
  - Move guests between systems to balance performance
  - Snapshots and cloning
- Hypervisor creates runs and manages guest OSes
  - Run in kernel mode
  - Implement device drivers
  - provide traditional OS services like CPU and memory management
- Examples: VMWare esx (dedicated) , Windows with Hyper-V (includes OS)

Colorado State University

Type 2 Hypervisors

- Run on top of host OS
- VMM is simply a process, managed by host OS
  - host doesn't know they are a VMM running guests
- poorer overall performance because can't take advantage of some HW features
- Host OS is just a regular one
  - Individuals could have Type 2 hypervisor (e.g. Virtualbox) on native host (perhaps windows), run one or more guests (perhaps Linux, MacOS)

Colorado State University

Full vs Para-virtualization

- Full virtualization: Guest OS is unaware of the hypervisor. It thinks it is running on bare metal.

- Para-virtualization: Guest OS is modified and optimized. It sees underlying hypervisor.
  - Introduced and developed by Xen
    - Modifications needed: Linux 1.36%, XP: 0.04% of code base
  - Does not need as much hardware support
  - allowed virtualization of older x86 CPUs without binary translation
  - Not used by Xen on newer processors

Colorado State University

CPU Scheduling

- One or more virtual CPUs (vCPUs) per guest
  - Can be adjusted throughout life of VM
- When enough CPUs for all guests
  - VMM can allocate dedicated CPUs, each guest much like native operating system managing its CPUs
- Usually not enough CPUs (CPU overcommitment)
  - VMM can use scheduling algorithms to allocate vCPUs
  - Some add fairness aspect
- Oversubscription of CPUs means guests may get CPU cycles they expect
  - Time-of-day clocks may be incorrect
  - Some VMMs provide application to run in each guest to fix time-of-day
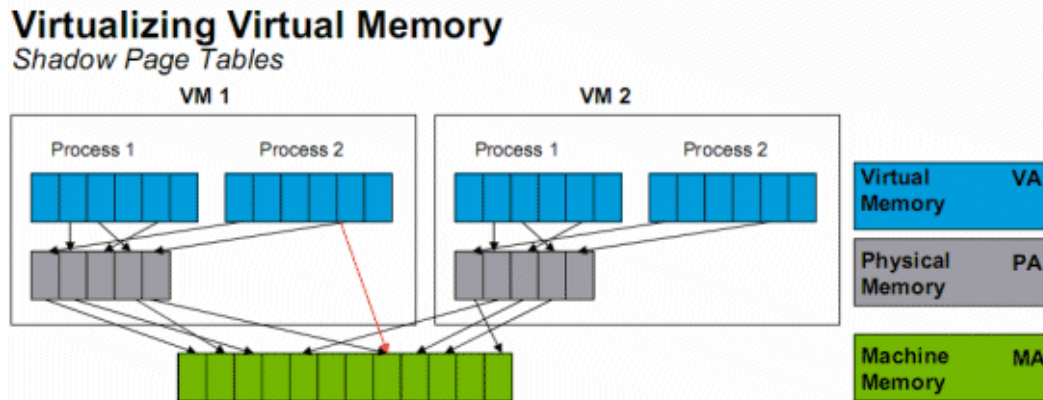
Colorado State University

Memory mapping:

- On a bare metal machine: OS uses page table/TLB to map Virtual page number (VPN) to Physical page number (PPN) (physical memory is shared). Each process has its own page table/TLB.

  – VPN -> PPN

- VMM: Real physical memory (*machine memory*) is shared by the OSs. Need to map PPN of each VM to MPN (Shadow page table)

  PPN ->MPN

Virtualizing Virtual Memory
Shadow Page Tables

VM 1		VM 2			
Process 1	Process 2	Process 1	Process 2	Virtual Memory	VA
				Physical Memory	PA
				Machine Memory	MA

Colorado State University

- VMM: Real physical memory (*machine memory*) is shared by the OSs. Need to map PPN of each VM to MPN (Shadow page table)

  PPN ->MPN

- Where is this done?

  – Has to be done by hypervisor ₜᵧₚₑ ₁. Guest OS knows nothing about MPN.

  – Page Table/TLB updates are trapped to VMM.

  It needs to do VPN->PPN ->MPN.

  – It can do VPN->MPN directly (VMware ESX)

Colorado State University

44

Virtual Machine (VM) as a software construct

- Each VM is configured with some number of processors, some amount of RAM, storage resources, and connectivity through the network ports.

- Once the VM is created it can be activated on like a physical server, loaded with an operating system and software solutions, and used just like a physical server.

- Unlike a physical server, VM only sees the resources it has been configured with, not all of the resources of the physical host itself.

- The hypervisor facilitates the translation and I/O between the virtual machine and the physical server.

Colorado State University
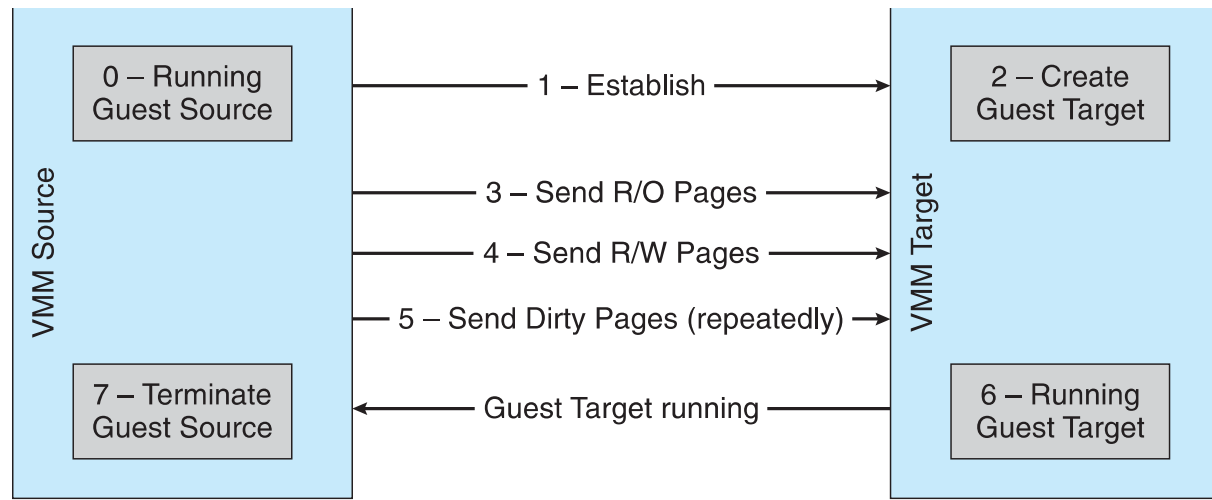
Virtual Machine (VM) as a set of files

- Configuration file describes the attributes of the virtual machine containing
  - server definition,
  - how many virtual processors (vCPUs)
  - how much RAM is allocated,
  - which I/O devices the VM has access to,
  - how many network interface cards (NICs) are in the virtual server
  - the storage that the VM can access
- When a virtual machine is instantiated, additional files are created for logging, for memory paging etc.
- Copying a VM produces not only a backup of the data but also a copy of the entire server, including the operating system, applications, and the hardware configuration itself

Colorado State University

Live Migration

Running guest can be moved between systems, without interrupting user access to the guest or its apps

- for resource management,
- maintenance downtime windows, etc

- Migration from source VMM to target VMM

  - Needs to migrate all pages gradually, without interrupting execution (details in next slide)

  - Eventually source VMM freezes guest, sends vCPU's final state, sends other state details, and tells target to start running the guest

  - Once target acknowledges that guest running, source terminates guest

Colorado State University

- Migration from source VMM to target VMM
  - Source establishes a connection with the target
  - Target creates a new guest
  - Source sends all read-only memory pages to target
  - Source starts sending all read-write pages
  - Source VMM freezes guest, sends final stuff,
  - Once target acknowledge that guest running, source terminates guest.

Colorado State University

- Developer can construct a virtual machine with
  - required OS, compiler, libraries, and application code
  - Freeze them as a unit … ready to run
- Customers get a complete working package
- Virtual appliances: "shrink-wrapped" virtual machines
- Amazon's EC2 cloud offers many pre-packaged virtual appliances examples of *Software as a service*

- *Question: do we really have to include a whole kernel in a shrink wrapped VM?*

Colorado State University

CS370 Operating Systems

Colorado State University

Yashwant K Malaiya

Fall 2025

Containers

Slides based on
- **Various sources**