Technology Overview

Cloud Concepts &Technologies

- Variety of technologies that make Cloud computing possible
- Virtualization
  - Key technology enabling Cloud Computing
  - Abstraction of an execution environment
  - Partitions physical resources into virtual resources
- Load Balancing
  - Help distribute workload across multiple machines to meet
- SDN
  - Separates control plane from the data plane
  - Creates agile network that are easier and cheaper to maintain.

Cloud Concepts &Technologies

- Scalability
  - Difficult to scale traditional IT infrastructure
    - Scale Up — Upgrade existing hardware (CPU, Memory, etc.)
    - Scale Out — Add more hardware
    - Not elastic
  - Cloud provide elasticity to accommodate for variability in workload.
    - Add or shrink resources based upon demand.

Cloud Concepts &Technologies

- Replication
  - Replication of data is important for business continuity
  - Cloud provides cheaper option to replicate data and application
    - Minimize capital expenditure
- Array-based replication
  - Automatically replicate data at the disk subsystem level
  - Independent of host and type of data being accessed.
  - Requires similar arrays at local and remote locations.

Cloud Concepts &Technologies

- Host-based replication
  - Runs on commodity servers
  - Agents installed on hosts communicate with each other
  - Replication can be block or file based.
  - Entire Virtual machine can be replicated in real-time.
- Network-based replication
  - An appliance sits on the network to transfer data (local to remote hosts)
  - Supports heterogeneous environment
  - Requires capital expenditure (hardware and software)

Rise of Virtualization

Virtualization is not new

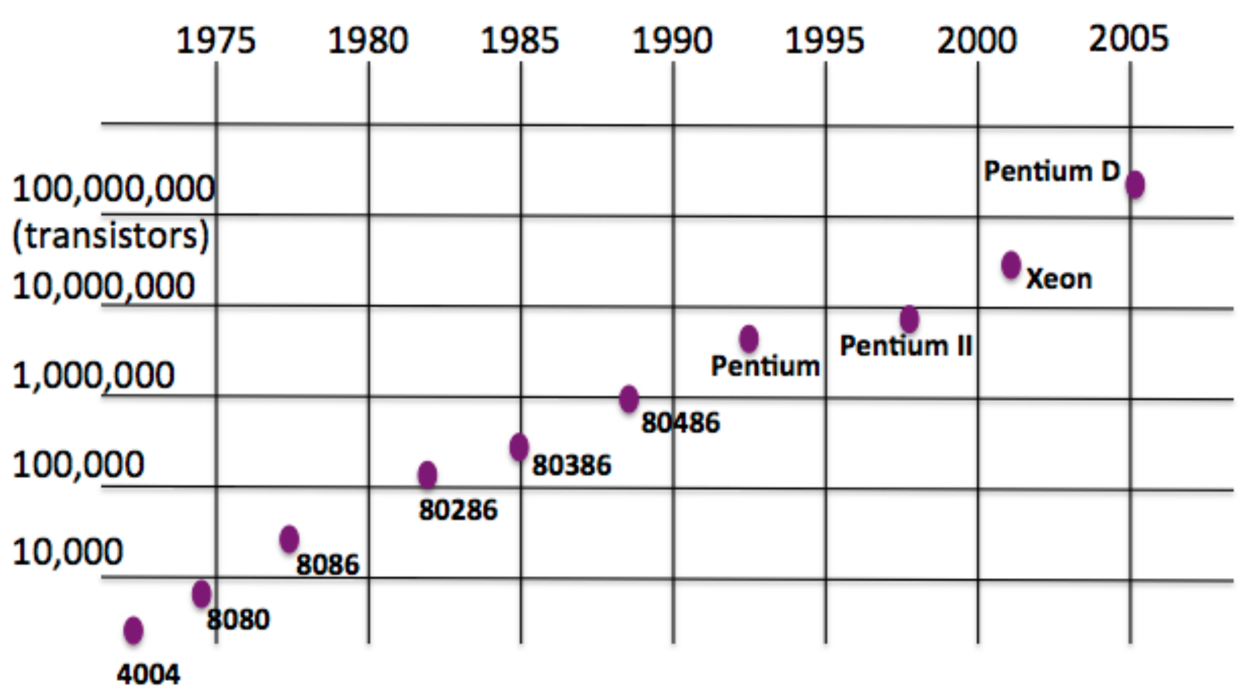
- Mainframe virtualization in the 1960s
- Codified in 1970s by Popek and Goldberg's three properties
- Fidelity – virtual environment should be identical to physical
- Isolation or Safety – VMM must have control of system resources
- Performance – little or no difference in performance
- An efficient VMM has all three properties

Rise of Windows

- Companies began using technology to achieve competitive advantages and save money (1970s)
- Proprietary solutions were expensive and inflexible
- Windows provided commodity platforms that drove down costs and defeated platform lock-in (1980s)
- Windows limitations often forced a ‘one server, one application’ policy.

Moore's Law

- Processing power doubles roughly every eighteen months.
- Originally, was coined around processing power.
- Today applies to many technologies



Rapid Data Center Growth

- Windows server growth drove datacenter growth
- Datacenter growth drove resource utilization
- Power, cooling, cables, square footage, staff, security
- Moore's Law made servers more powerful, but less efficient due to application deployment practices

Trends that accelerated virtualization

- Consolidation
- Running multiple workloads on a single host
- Containment
- Faster server provisioning
- Dynamic Load Balancing
- Faster development and test environment.
- OS Independence (Reduce vendor lock-in)

OS and Application Virtualization

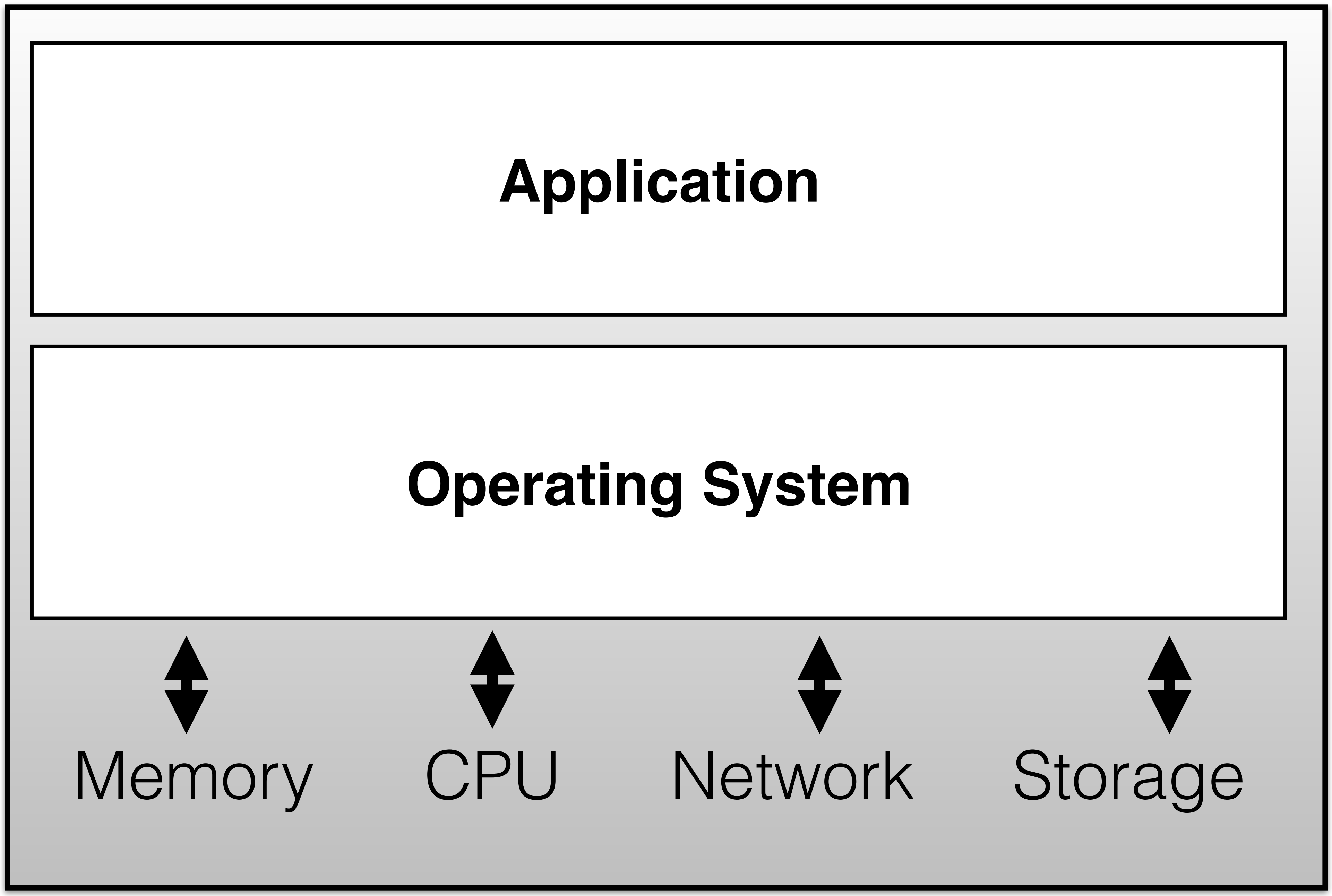
OS Virtualization

- Virtual Workspaces — An abstraction of an execution environment
- Virtualization decouples the application and operating system from HW
  - Allows consolidation
  - Enhances utilization
  - Replicated, moved, suspended quickly

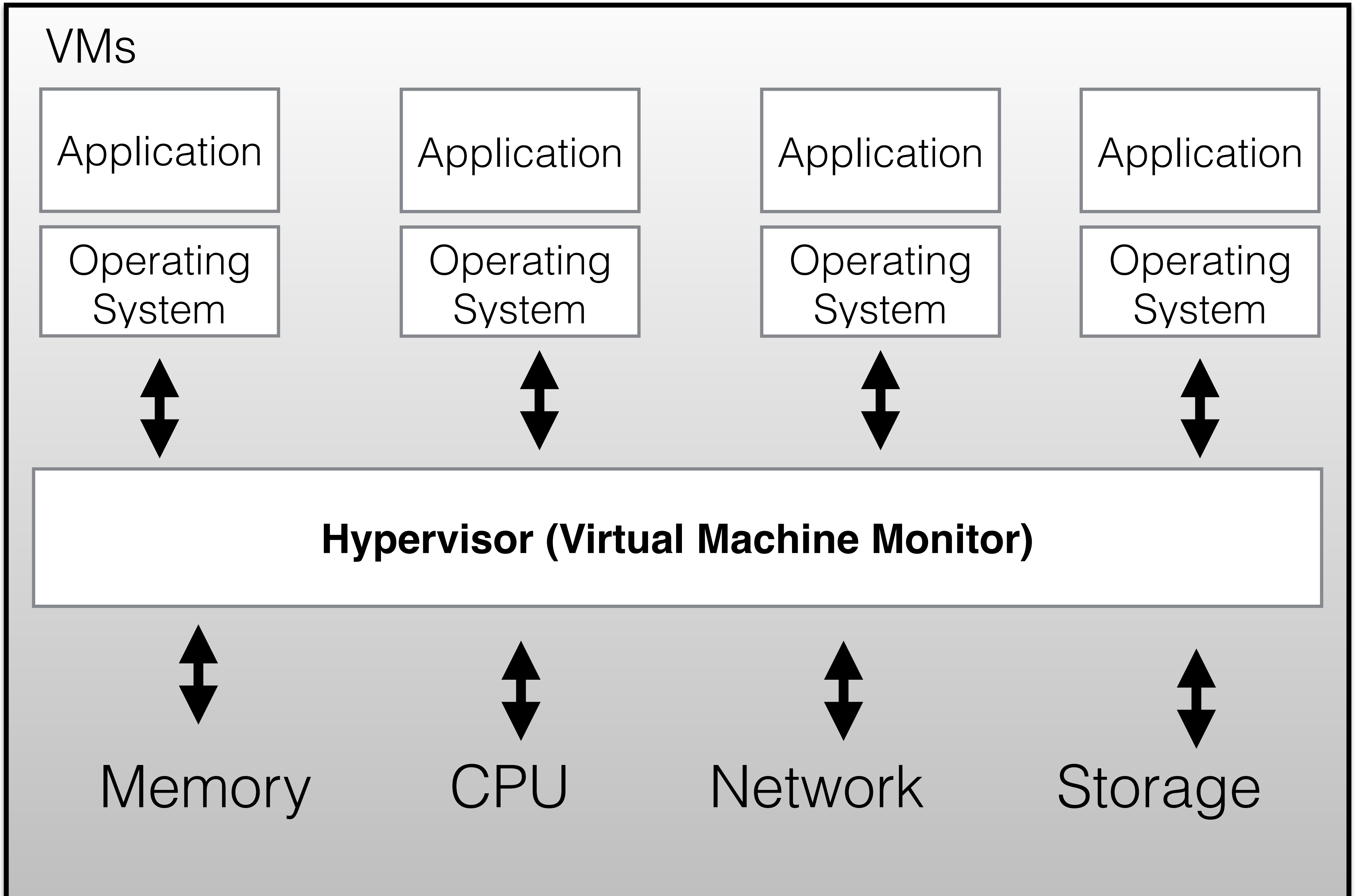
OS Virtualization

- Allows same physical host to serve different workload and isolate each workload.
- Host OS runs on the host, with VMs (workloads) running on top
- These workload can run different OS
- Process isolation is provided by Kernel Host
- Each process have their own file system, processes memory, devices, etc.

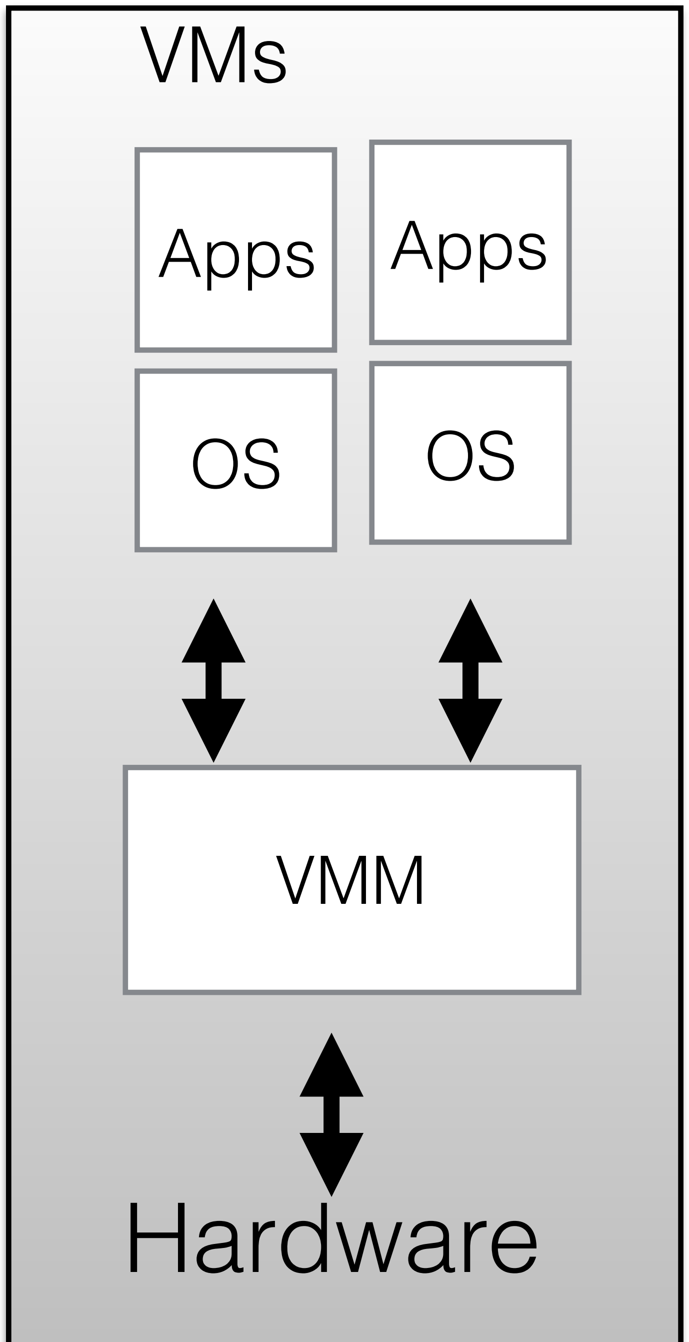
Dedicated Server



Virtualized Server



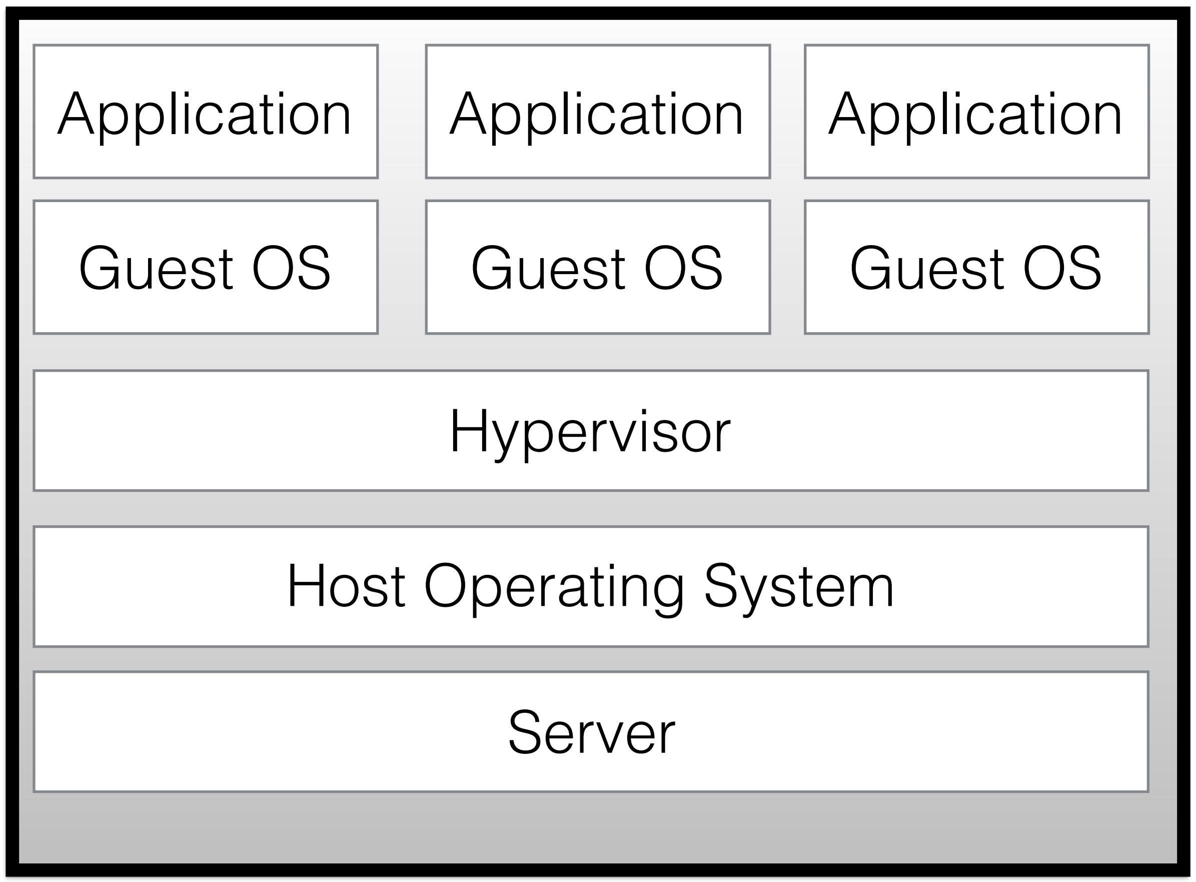
Virtualization



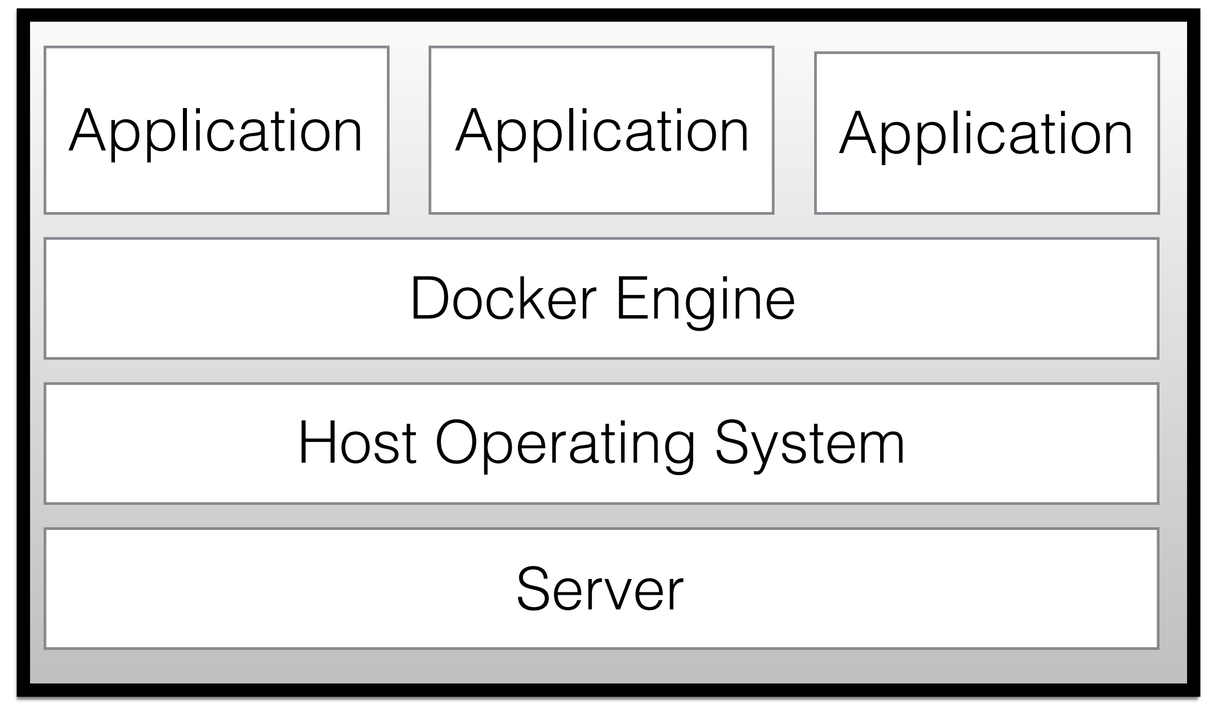
- Hypervisor (VMM) abstracts all hardware resources
- Guest OS runs in user mode
- VMM runs in kernel mode
- VMM responsible for controlling physical platform resources and I/O mapping
- Advantage: Run multiple OSs on the same physical platform
- VMM allocate resources based upon the request.

Application Virtualization

Virtualized Servers



Containers (Docker)



Container

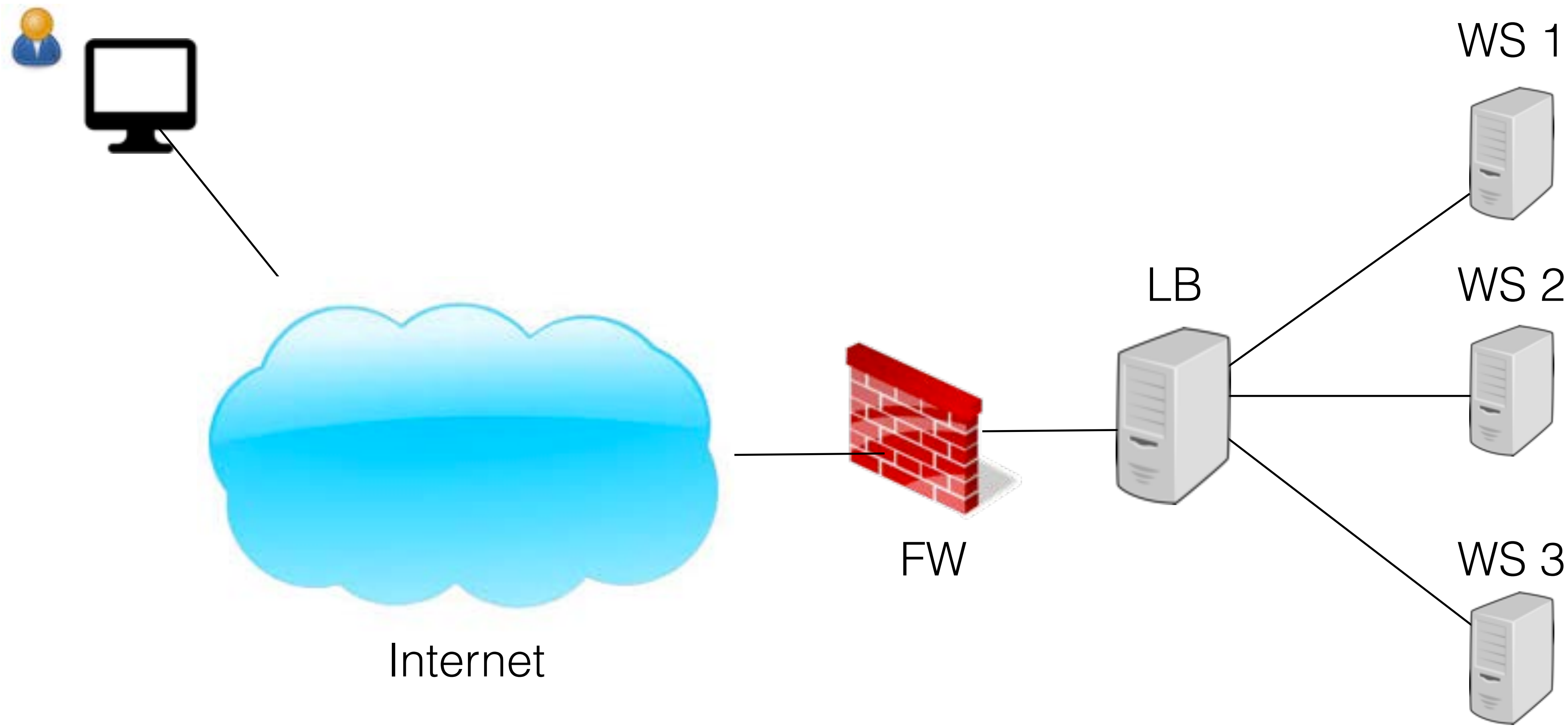
- Container abstracts OS Kernel
  - Hypervisor on abstracts the entire device.
- Container makes it easier to package and move program into different cloud environment
- Container uses shared operating system
  - VM provides more isolation with guaranteed resources

Load Balancer

Load Balancing

- Important characteristics of cloud is scalability
- Cloud computing resources can scale on demand
- Load Balancer distributes workload across multiple servers
- Helps achieve optimum utilization of resources
  - Achieve high availability and reliability.
  - In the even of a resource failure, the LB can reroute traffic

Deployment



Algorithms

- Load balancer can be programmed to distribute traffic in a variety of ways.
- Following are some of the algorithms (not exhaustive)
- Round Robin
  - Servers are selected one by one to serve the incoming requests
  - Each server gets a request in a circular fashion
  - All servers have same priority.
- Weighted Round Robin
  - Servers are assigned some weight
  - Incoming requested are directed proportion to the weight.
  - Each server will not receive same number of requests.

Algorithms

- Low Latency
  - Load balancer monitors the latency of each server
  - Incoming requests are routed to server with the lowest latency
- Priority
  - Each server is assigned a priority
  - Incoming request is routed to the highest priority server that is available
  - Lower priority server gets traffic when high priority servers are busy

Implementation

- Can be implemented in hardware or software
- Software based LB runs on OS
    - Easily run virtualized
- Hardware based solutions use specialized hardware to distribute traffic.

Hypervisor & Virtual Resources

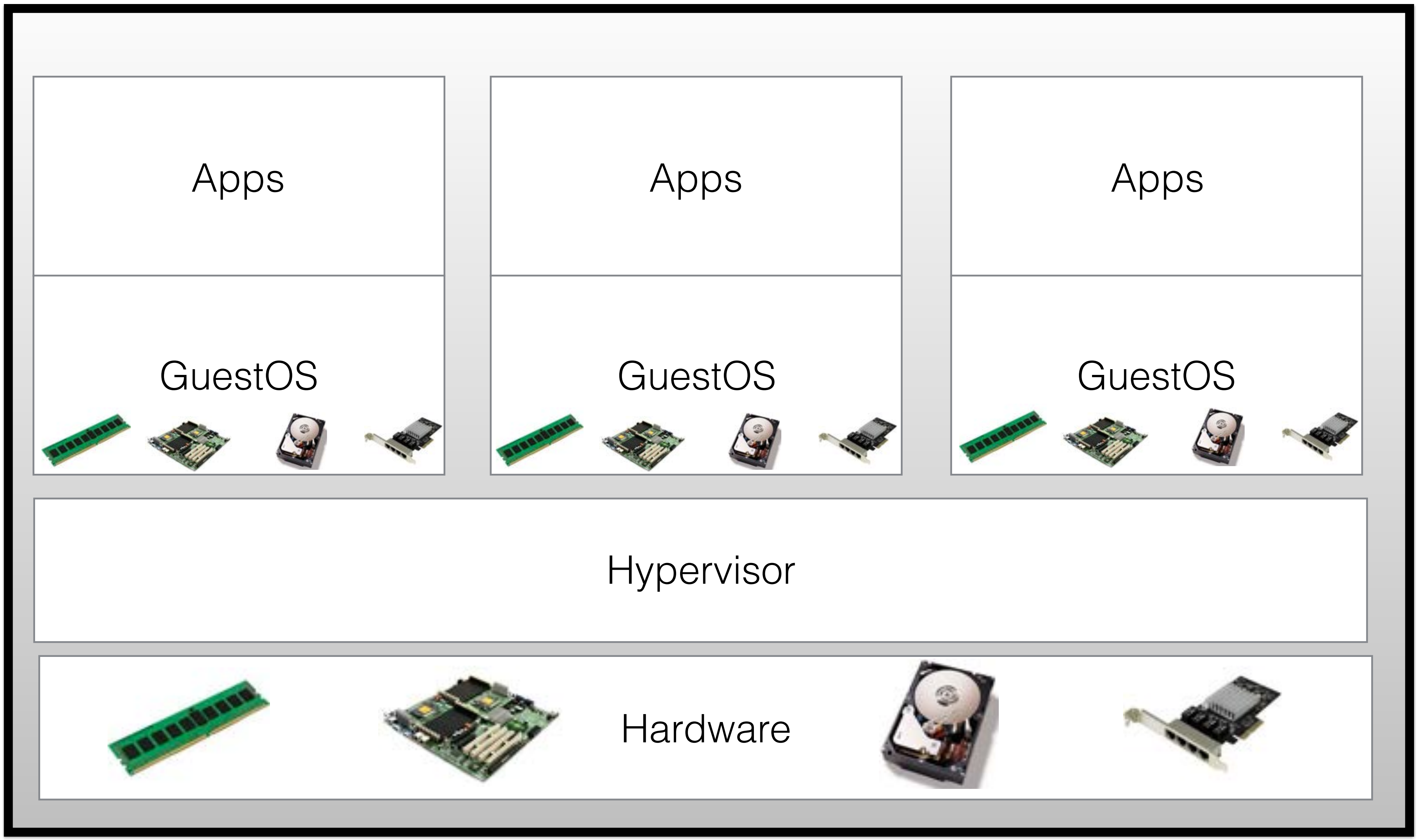
Role of Hypervisor

- Arbitrator of resources
- Sits between the physical and virtual resources
- Allocates resources
- Provides virtual environment for workloads
- Without hypervisor OS communicates directly with hardware resources
- Without hypervisor multiple OS would like to access hardware resources simultaneously.

Traffic Cop

- Gives illusion to the Guest OS that it has direct access to hardware resources
  - Use disk drives, Memory, Network, etc.
- Balance the workload
  - Each guest makes constant demands
  - Hypervisor provides timely response to each request
    - Ensuring adequate resources for all virtual machines

Hardware Abstraction

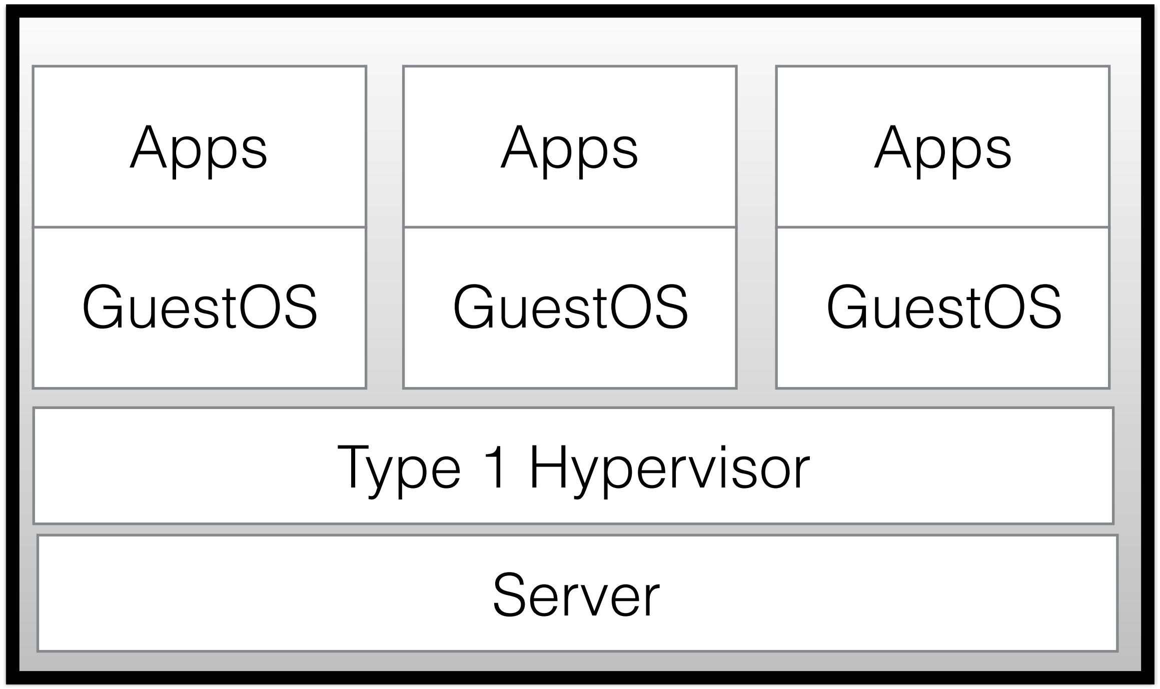


Types

- Two classes of hypervisors:
  - Type 1
  - Type 2

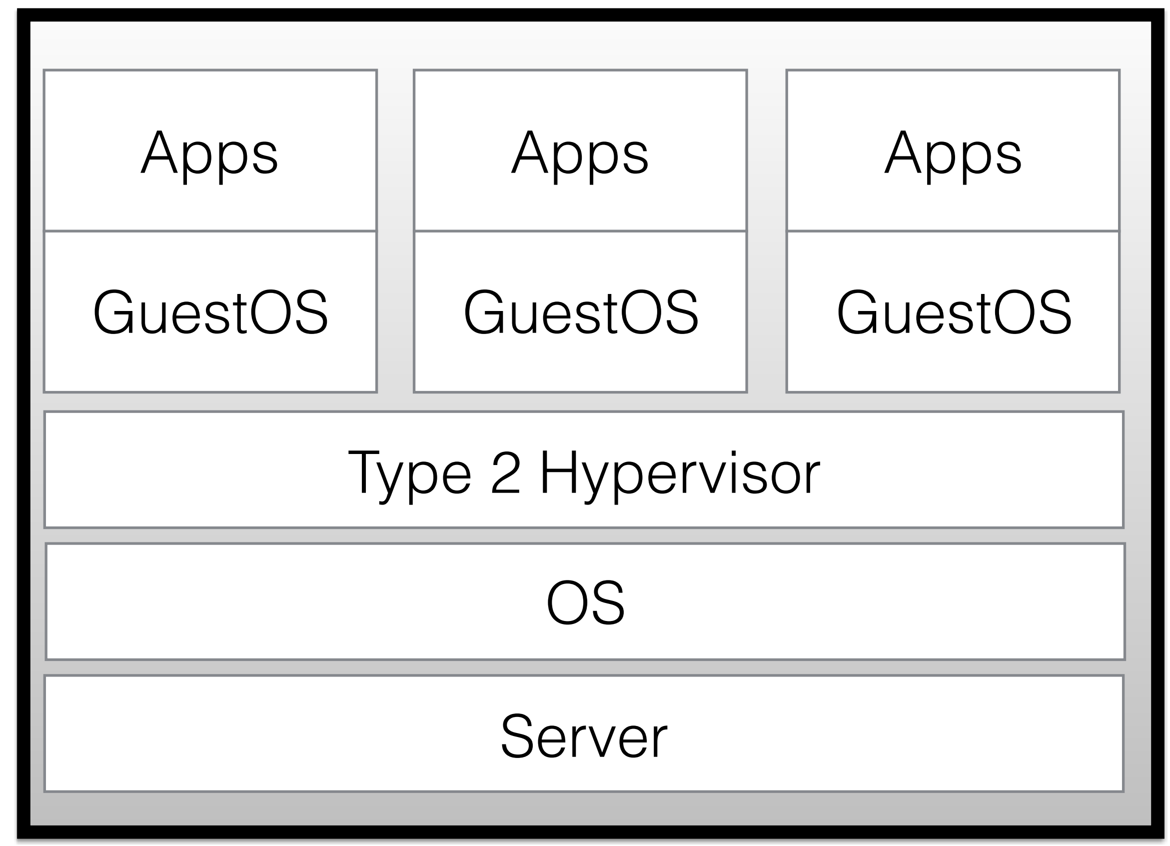
Type 1

- Runs directly on the hardware
- More secure and more available
- Offers better performance to the guests
- Generates less overhead
    - No intermediate layer of OS
    - More Virtual machines can run on the box



Type 2

- Runs like an application on OS
- Can support large range of hardware — OS dependent
- Not as efficient as Type 1
    - Extra layer between itself and the hardware
    - Every operation from guest OS travels from Hypervisor to OS to Hardware
- Less reliable — Underlying OS can impact the hypervisor and VMs
- Good for desktop development environment
    - Single developer working on multiple VMs.



VMWare ESX (Type 1)

- First commercially available hypervisor (1998) for the x86 platform
- Currently market leader in user share and maturity of offerings
- Architecture not tied to an operating system
- Initial deployment included:
  - Hypervisor
  - Service Console
    - Linux-based console sits alongside the Hypervisor
    - Acted as management interface to Hypervisor
- Subsequently removed Service console — Security & Size
  - ESXi

Xen (Type 1)

- Began as a Cambridge University research project
- First released in 2002 as an open source project
- Currently exists as an open source project
- The core has been used by a number of vendors including Citrix and Oracle
- Implementation different from VMWare architecture
- Special guest — Domain 0 (Dom0)
  - Guest is booted when hypervisor is booted
  - Has special admin privileges

Microsoft Hyper-V (Type 1)

- First release in 2008, but virtualization was available through Virtual Server in 2005
- Different nomenclature
  - Virtualized workloads are called partitions
- Similar architecture to Xen
  - Device drivers part of the parent partition (similar to Dom0)
  - Like Dom0 parent partition runs an OS
- Parent partition is based on Microsoft Windows
  - Creates and manage child partitions and handle management functions

Other Solutions

- Red Hat KVM
- Initially XEN, now kernel based (KVM)
- Oracle VM is XEN based
- Many other commercial and free solutions available but the space is still evolving

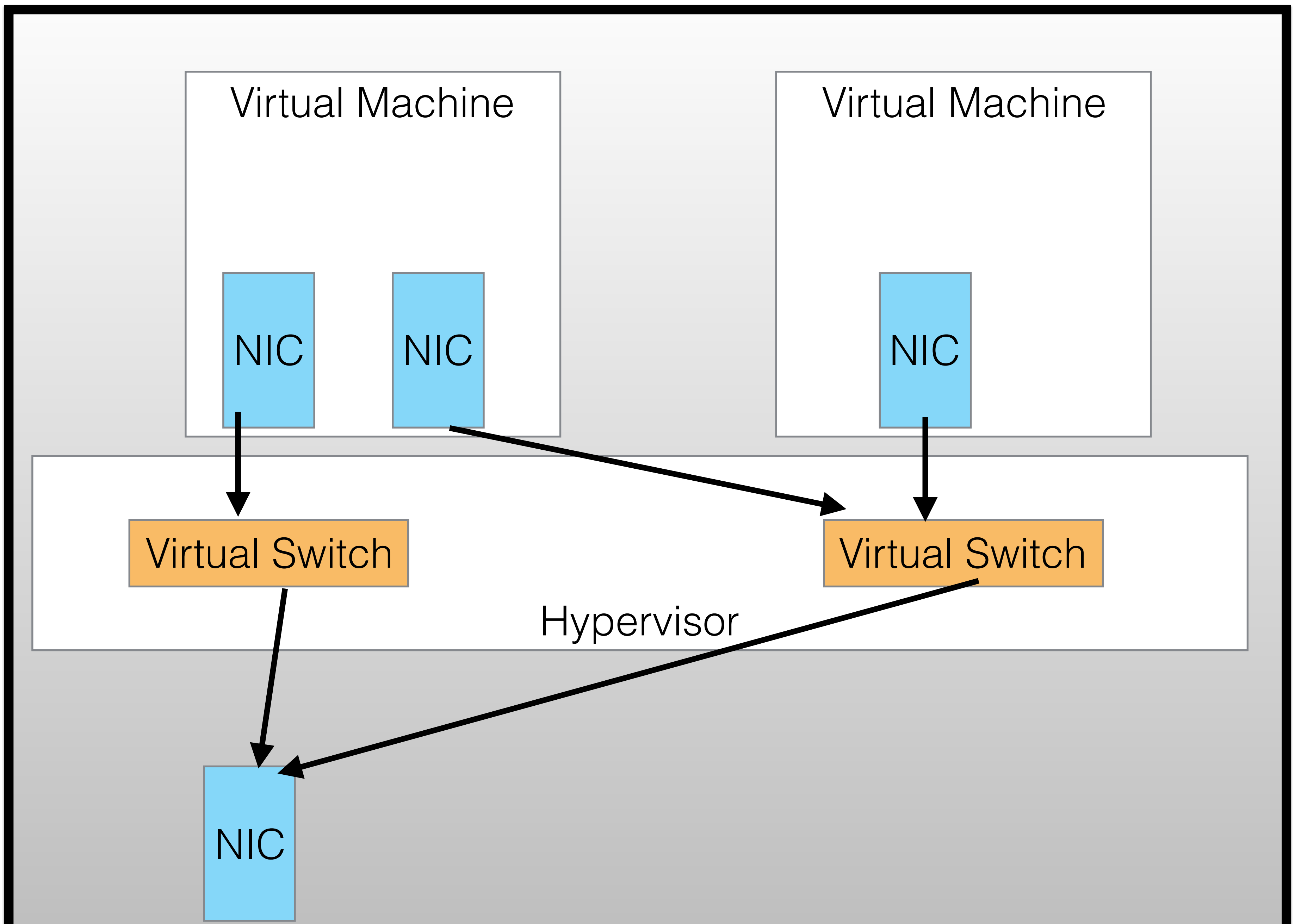
Virtual Machines (VM)

- Has many characteristics of a physical server
  - Supports OS
  - Configured with a set of resources
- VM is nothing more than a set of files
  - Configuration File
    - describes resources that the VM can utilize.
  - Virtual disk files

Virtual Resources

- Virtual CPUs (vCPUs) are scheduled slices on Physical CPUs (pCPUs)
- A processor core is equivalent to a vCPU
- A VM's memory is mapped to physical memory but managed by the hypervisor
- A VM can only see and utilize the amount of memory it has been allocated.
- Like vCPUs, you can only adjust the amount of allocated memory
- There are many memory optimization features available.
- The resource most systems run out of first

Virtual Networking



Virtual Storage

- VM talks to virtual SCSI driver
- Hypervisor passes data blocks to/from physical storage
- VM don't have to worry about how they are connected to the storage device
    - Fibre channel, iSCSI, NFS, etc.

Storage

Background

- Data storage is an ever-expanding resource.
- Making data store very pervasive
- Everything from refrigerators to automobiles now contains some amount of data storage.
- Appliances like GPS, DVR — part of daily routine
- PCs, smart phones, music players, and tablets have experienced storage growth as each new generation of the devices
- Also true in traditional DCs — data far greater now (types of data)

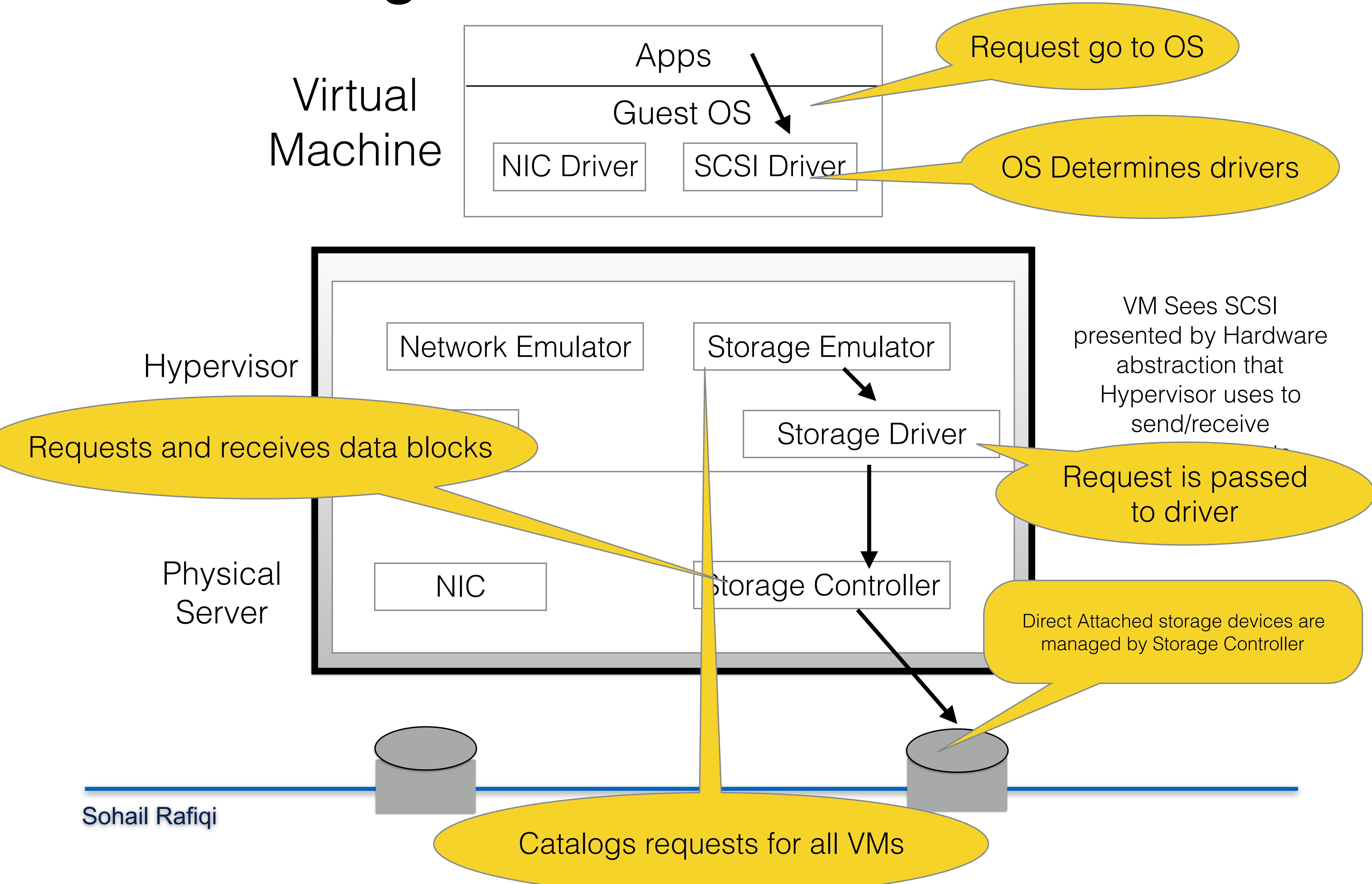
Background

- 2008 — 8 Exabyte (8 quintillion bytes) data generated
- UC Berkeley concluded — Data would double every 6 months
- 2010 — Passed 1 zettabytes
- 2011 — According to IDC generated 2 zettabytes
- 2020 — Forecast to have 40 zettabytes annually

Traditional environment?

- Most computing devices request information in a similar fashion
- OS controls the access to various I/O devices
- Application program request OS for information to process
- OS passes request to the storage subsystem
- Subsystem passes information (data blocks)
- OS transfers data blocks to the applications

Storage in Virtual Environment



SAN/NAS

- Key portion of virtualization storage architecture is clustering and shared storage

- SAN or NAS allows server to access disk storage that is not part of its hardware configuration

- Can be used both by virtual or physical servers

    - Making migration to virtual environment smoother

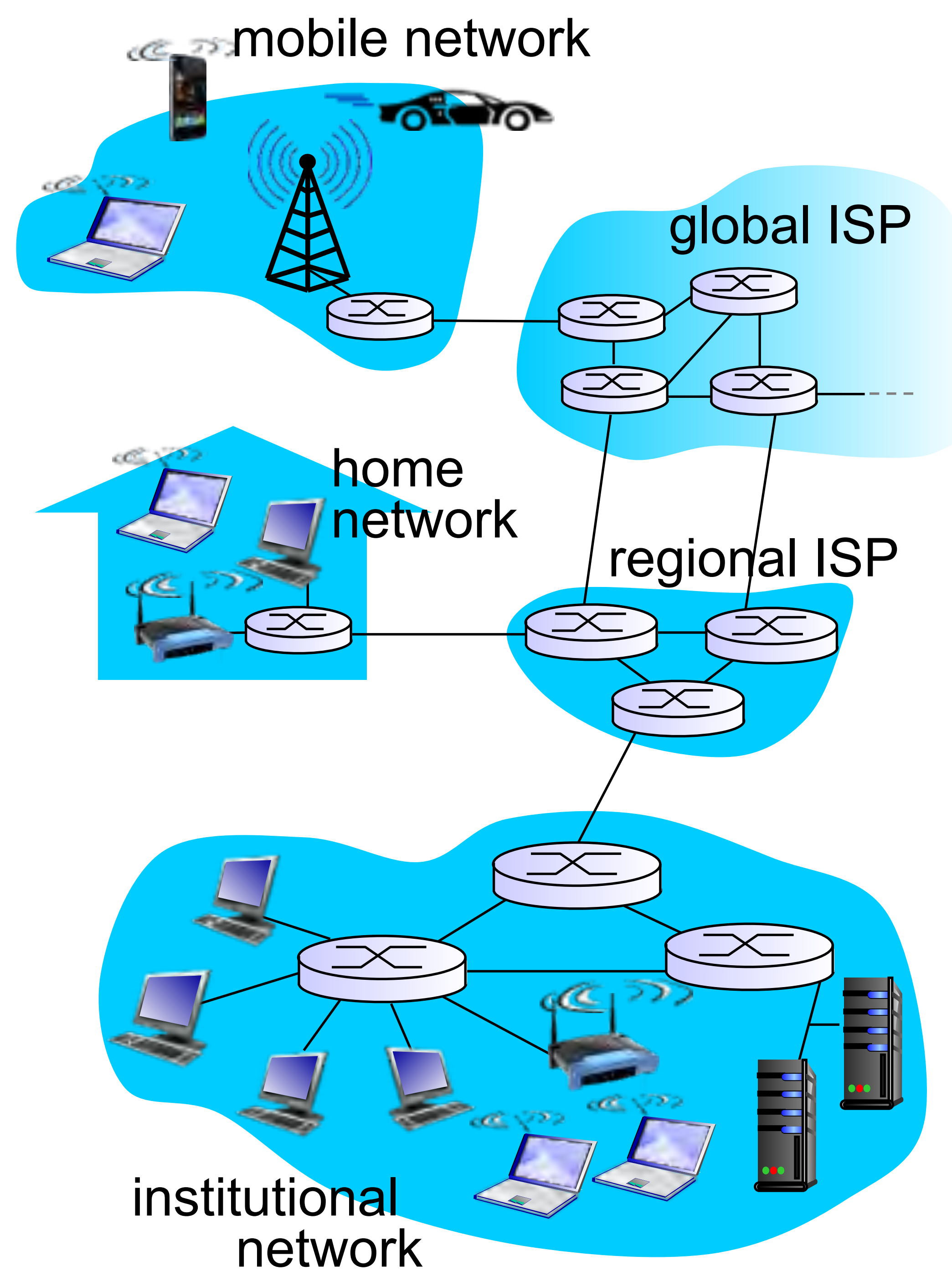
Summary

- Storage virtualization is a technical means to what are essentially business ends.
- Reducing the cost of data storage administration,
- Maximizing utilization of storage assets,
- Dynamically aligning data storage capacity to changing application requirements,
- Ensuring high availability access to data, and
- Safeguarding an organization's information

Network Overview

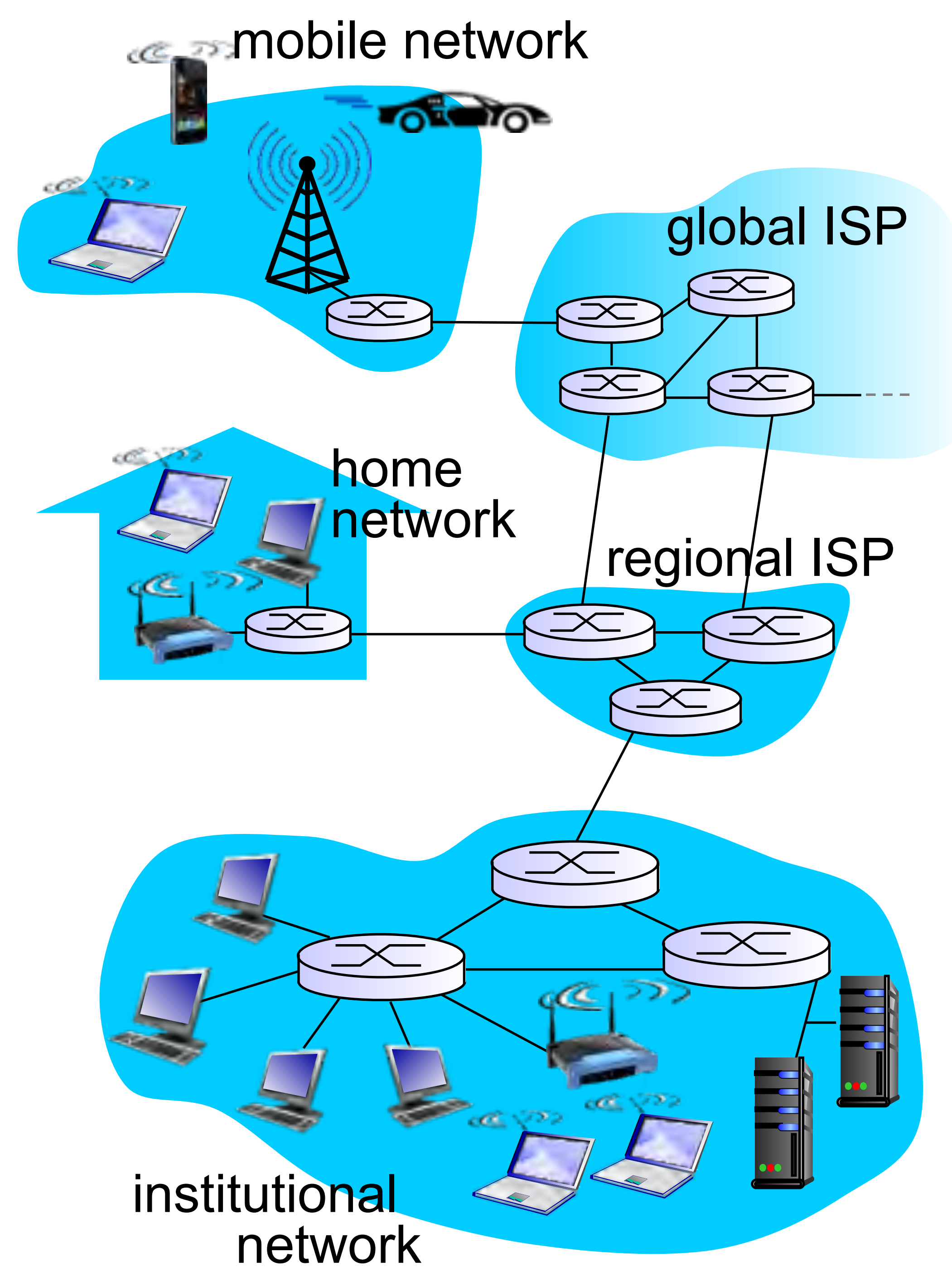
What's the Internet: "nuts and bolts" view

- Internet: "network of networks"
  - Interconnected ISPs
- protocols control sending, receiving of msgs
  - e.g., TCP, IP, HTTP, Skype, 802.11
- Internet standards
  - RFC: Request for comments
  - IETF: Internet Engineering Task Force



What's the Internet: a service view

- Infrastructure that provides services to applications:
    - Web, VoIP, email, games, e-commerce, social nets, …
- provides programming interface to apps
    - hooks that allow sending and receiving app programs to "connect" to Internet
    - provides service options, analogous to postal service

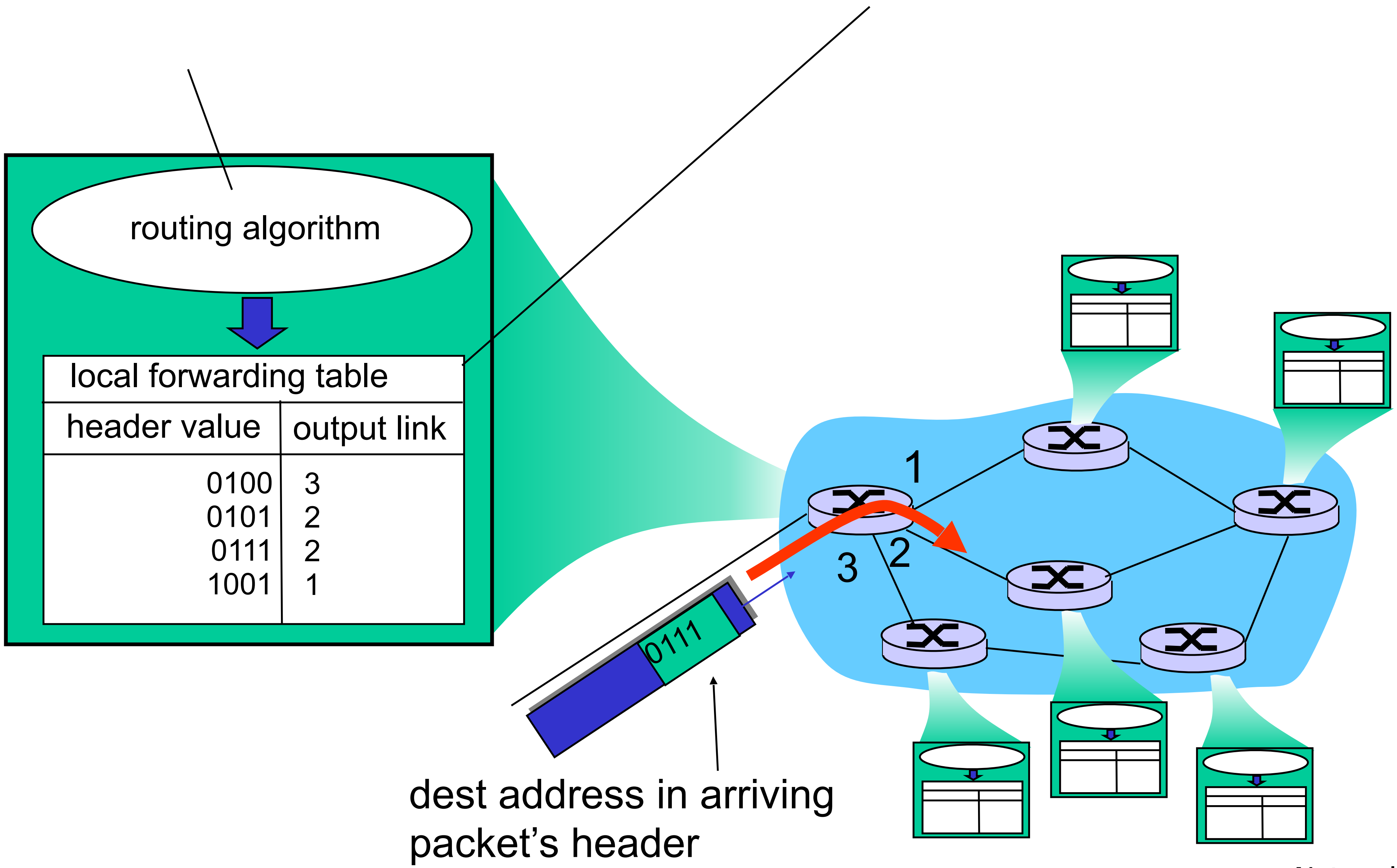


Two key network-core functions

routing*:* determines source-destination route taken by packets

- *routing algorithms*

forwarding*:* move packets from router's input to appropriate router output



Protocol "layers"

Networks are complex,
with many "pieces":

- hosts
- routers
- links of various media
- applications
- protocols
- hardware, software

*Question:*_

is there any hope of *organizing* structure of network?

…. or at least our discussion of networks?

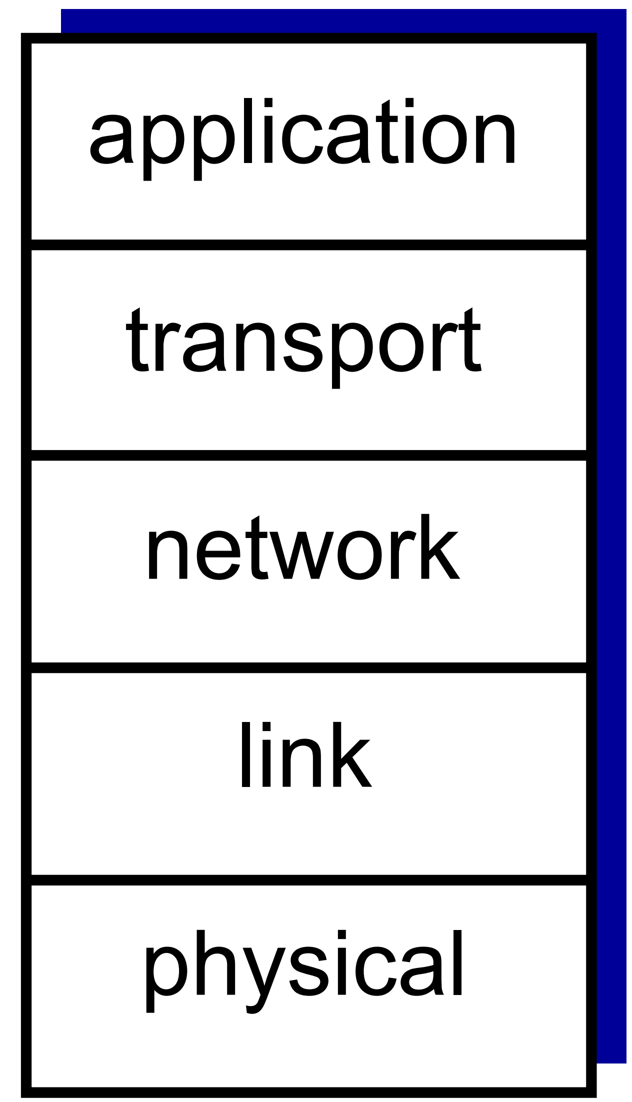
Why layering?

dealing with complex systems:

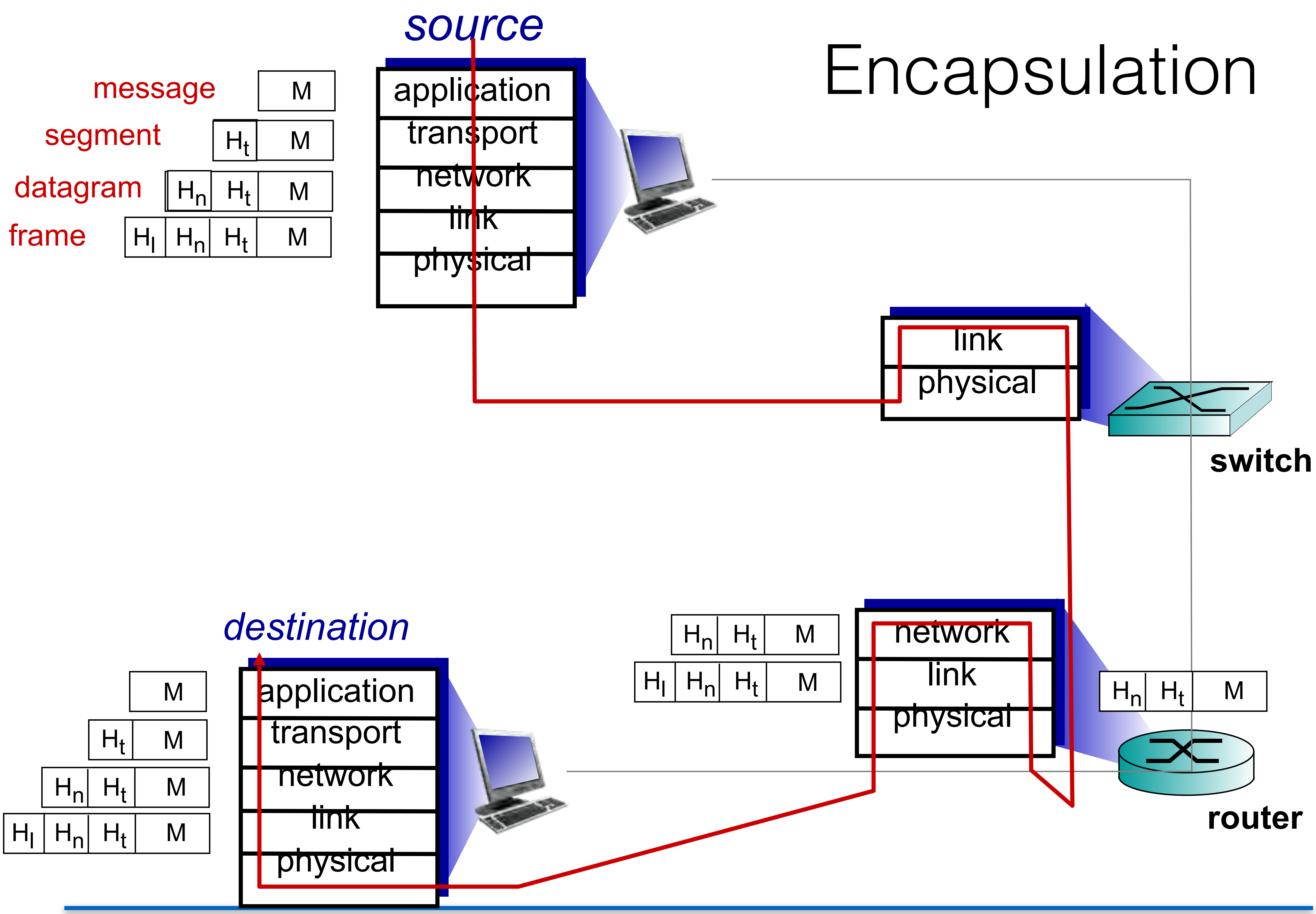
- explicit structure allows identification, relationship of complex system's pieces
  - layered *reference model* for discussion
- modularization eases maintenance, updating of system
  - change of implementation of layer's service transparent to rest of system
  - e.g., change in gate procedure doesn't affect rest of system

Internet protocol stack

- *application:* supporting network applications
  - FTP, SMTP, HTTP
- *transport:* process-process data transfer
  - TCP, UDP
- *network:* routing of datagrams from source to destination
  - IP, routing protocols
- *link:* data transfer between neighboring network elements
  - Ethernet, 802.111 (WiFi), PPP
- *physical:* bits “on the wire”

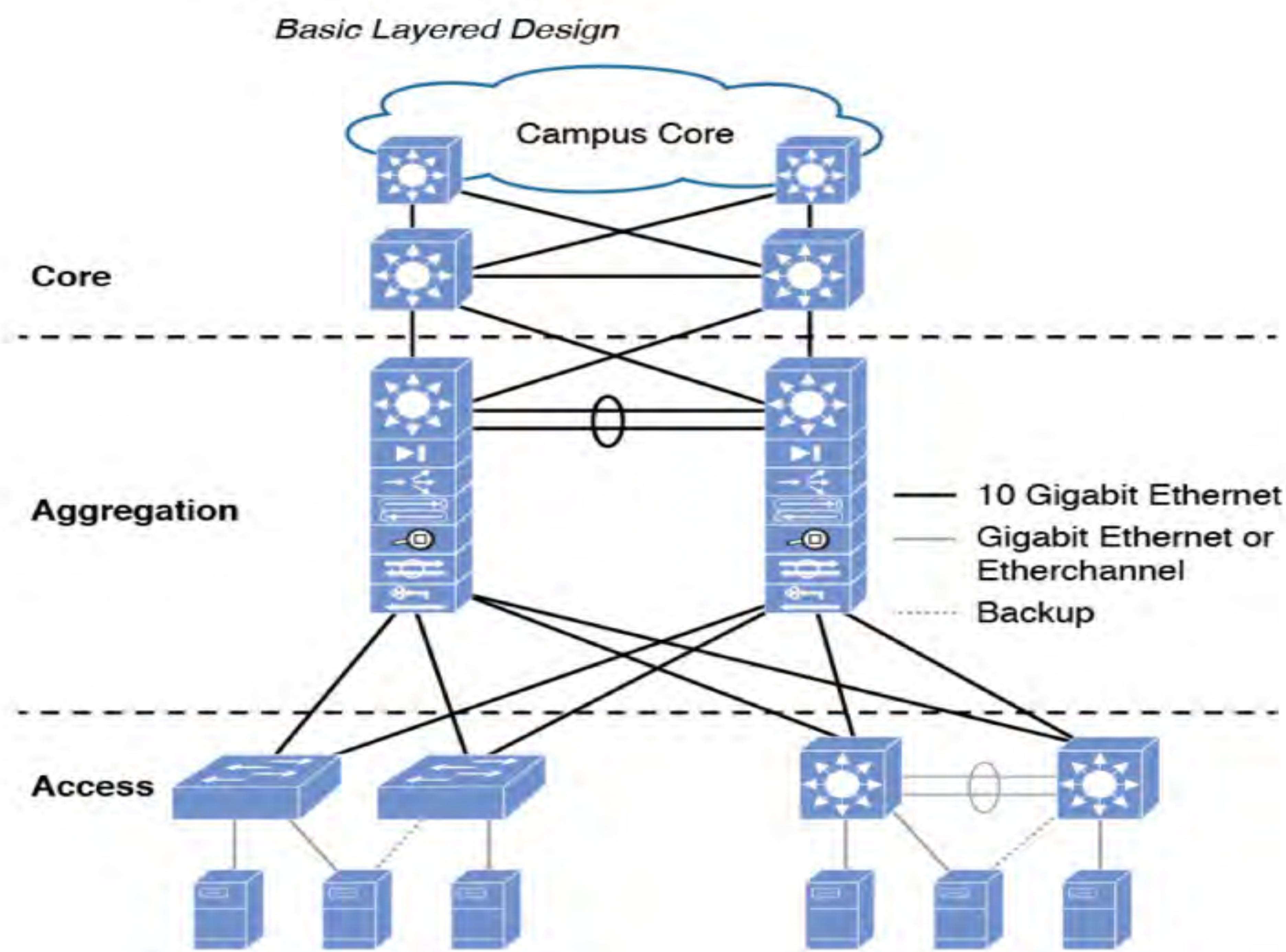


Encapsulation



Network Virtualization

Traditional Data Centers



Addressing

- The key to all of this is the addressing scheme.
- The Layer 3 or IP address is used to get the packet across the wide-area network to the right data center
- Layer 2 address tells all the switches in the data center which server the traffic should be sent to.
- In the preceding scenario, the following was true:
  - The application was associated with a single server,
  - and all the application-based addressing and programming of the network was based on where that physical server was located,
- Server were dependent on a layer 3 routing subnet (location dependent)
  - Server's IP address will route to the correct data center's access switch

Addressing with Virtual Machines

- First is that VMs do not roll off a factory line.
  - They get created
- VM machines move a lot
  - As a result, we have some new problems to solve.
    - First, who or what creates MAC addresses, and
    - Second, how do we account for all this moving around because the rest of the network has to know where to send traffic

Addressing with Virtual Machines

- VM software such as VMSphere or Citrix provides
  - Unique MAC address for each VM created.
  - These VM managers also assign a virtual NIC (vNIC) or multiple vNICs
    - NIC is a specific piece of equipment within a device that uses the MAC address.
- Most of these VM managers also enable you to manually configure the MAC
  - By assigning each individual VM its own MAC
    - You can address that VM individually on the network
    - VM MAC is independent of the physical server’s network card
    - Capable and free to migrate to another server without any restrictions

Network Virtualization

- Similar to server virtualization
- Abstraction of the network endpoints from the physical arrangement of the network.
- Network Virtualization refers to the creation of logical groupings of endpoints on a network.
- Endpoints are abstracted from their physical locations
  - VMs can look, behave, and be managed as if they are all on the same physical segment of the network.

Network Virtualization

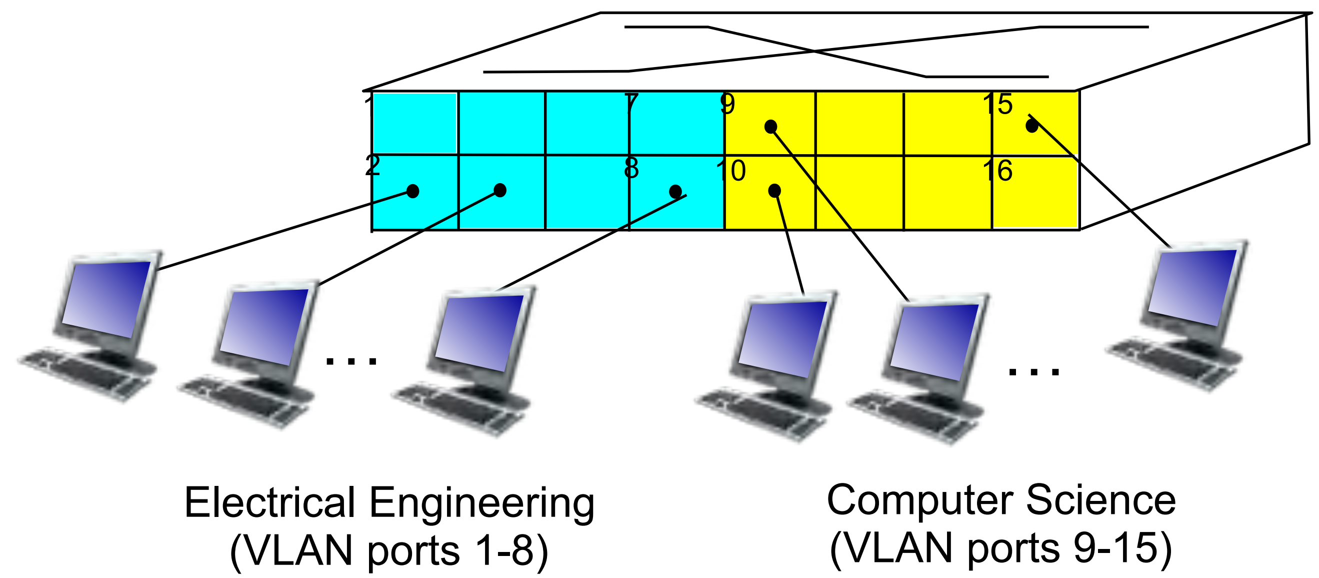
- Not new
  - VLAN, VPN, MPLS
    - Group physically separate endpoints into logical groups.
    - Enhances efficiencies in traffic control, security, and network management.
- Network is virtualized to get VM mobility
- What is new here:
  - Automation and management tools that have been purposely built for the scale and elasticity of virtualized data centers and clouds.

VLANs

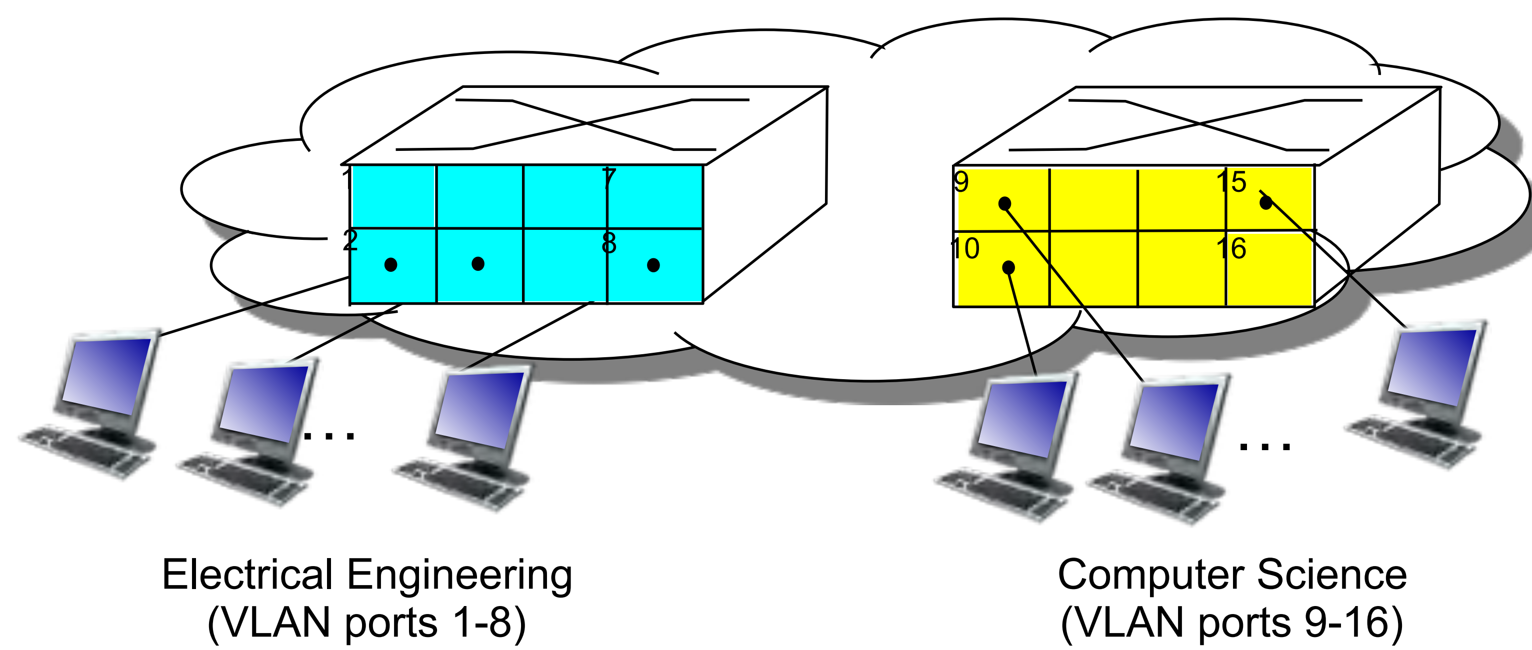
port-based VLAN: switch ports grouped (by switch management software) so that single physical switch ……

Virtual Local Area Network

switch(es) supporting VLAN capabilities can be configured to define multiple <u>virtual</u> LANS over single physical LAN infrastructure.

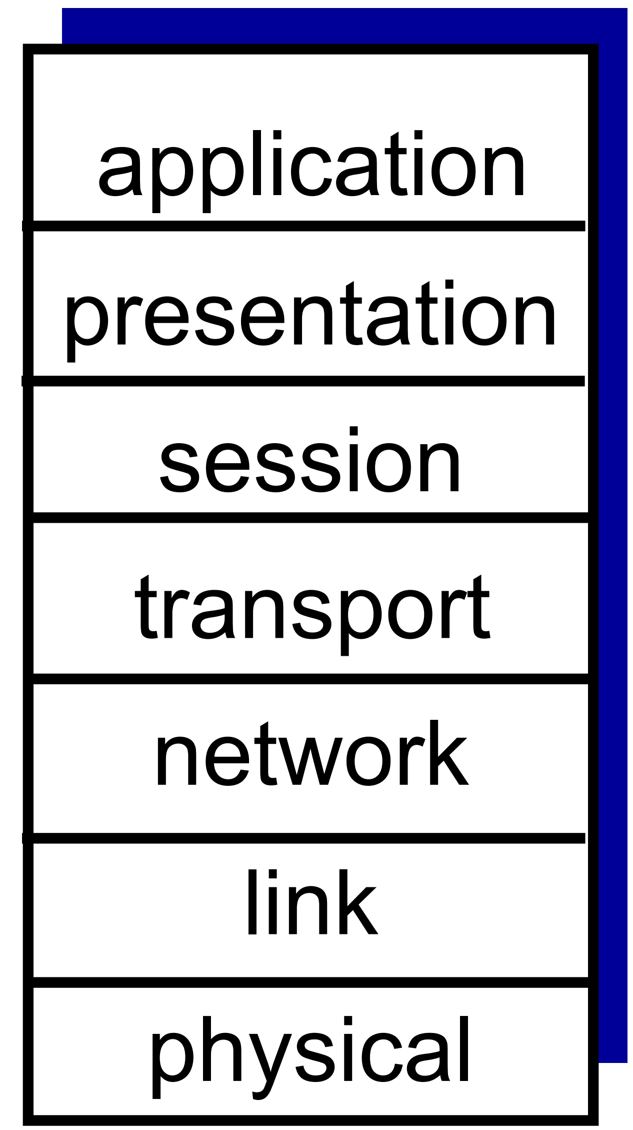


… operates as multiple virtual switches



Network Functional Virtualization (NFV)

- NFV refers to the virtualization of Layer 4 through 7 services
- Basically, this is converting certain types of network appliances into VMs,
    - which can then be quickly and easily deployed where they are needed.
- NFV came about because of the inefficiencies that were created by virtualization.
- Virtualization causes a lot of problems, too.
    - One of them was the routing of traffic to and from network appliances
    - With VMs springing up and being moved all over, the traffic flows became highly varied
    - Cause problems for fixed appliances that had to serve the traffic.
- NFV allows to create a virtual instance of network function (Firewall, Load Bal)
    - Can be easily “spun up” and placed where it is needed, just as they would a VM.



Virtualizing the network

- Network virtualization allows users to fully realize server virtualization features:
  - vMotion, snapshot backups, and push button disaster recovery (to name just a few).
  - The most common reason for virtualizing the network is precisely to get VM mobility

Summary

- Good old technique that has been around for many years
  - Makes server virtualization, and connecting VMs, easier and efficient.
  - It's easy to see why when you imagine the VMs being spun up here, there, and everywhere in a virtualized data center or cloud and
  - then being paused, moved, started again, or even being moved while still being active.
- With all that spontaneous creation without any regard for the specific physical location in the data center
  - or even with regard to a specific data center
  - having the ability to create and manage logical groupings becomes critical.