Introduction to Cloud Computing and Virtualization

Gabriel Scalosub

Introduction to Cloud Computing and Virtualization

Gabriel Scalosub

Borrowed extensively from:

Elisha Rozensweig, Erez Biton, Roy Campbell, Reza Farivar, Niv Gilboa, James Mickens

and various other papers/resources (see list at the end)

Outline

- Syllabus and Course Administration
- What is Cloud Computing

Outline

- Syllabus and Course Administration
- What is Cloud Computing

Syllabus

- Available on the course website on Moodle
  - Staff
    - Lecture, recitation, contact details
  - Course requirements
  - Bibliography and further reading
    - Provided in each lecture
  - Topics (coming up next)

(Tentative) Topics

(order might be somewhat different…)

- Introduction to cloud computing and virtualization
  - Including VMs & containers
- Cloud and workload orchestration and management
  - Including Kubernetes & OpenStack
- Cloud Networking
  - Including SDN & NFV
- VM Placement
- Cloud Storage
- Programming in the cloud
  - Including MapReduce
- Cloud Security
- Applications in the cloud
  - Including 5G, Big Data, ML

- Meta-topics
  - Industry “de-facto” standards
  - Hands-on
  - DevOps
  - Theory
  - Introduction…

Outline

- Syllabus and Course Administration
- What is Cloud Computing

What is Cloud Computing, “Officially”?

Mell and Grance, “The NIST Definition of Cloud Computing”, NIST Special Publication 800-145, 2011

"*Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g. networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction*".

Thank you!

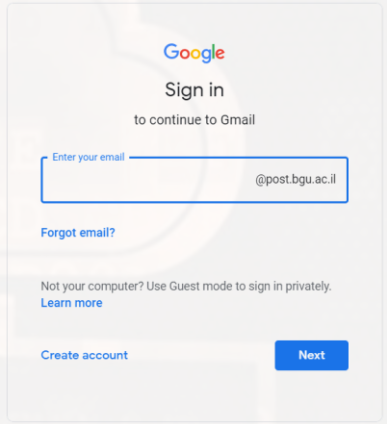
And see you next week??

Outline

- Syllabus and Course Administration
- What is Cloud Computing
- What is Virtualization
- Cloud Miscellany

Use Cases

- Giants:
  - Search
  - Email
  - Online retail
  - Social networks
  - Video distribution (and other CDNs)
  - Commodity software (e.g., Office365)
- SMEs & Startups
  - FrontEnd (towards clients)
  - BackEnd (“behind the scenes”)
    - Big Data, ML, DBs, services



amazon

facebook

NETFLIX

Office 365

Use Cases

- Giants:
  - Search
  - Email
  - Online retail
  - Social networks
  - Video distribution (and other CDNs)
  - Commodity software (e.g., Office365)
- SMEs & Startups
  - FrontEnd (towards clients)
  - BackEnd (“behind the scenes”)
    - Big Data, ML, DBs, services

- Huge amounts of data
- Fault tolerance and redundancy
- Strict SLAs (delay, BW, availability)
- Dynamic data & infrastructure
- Used by everyone
  - And their grandparents…

Pre-cloud Corporate IT: “Everything”

- Corporate owns everything: OnPrem
  - Infrastructure, data
  - “secure”, internal
- Corporate is responsible for everything
  - Downtime, upgrades, scale, personnel, …
- Corporate pays for everything
  - CapEx (Captial Expenditure – **הוצאות הון**)
    - Infrastructure
  - OpEx (Operating Expense – **הוצאות תפעוליות**)
    - Energy, cooling, personnel (IT teams), maintenance
  - Inefficiency
    - Low utilization / overcapacity, high energy costs

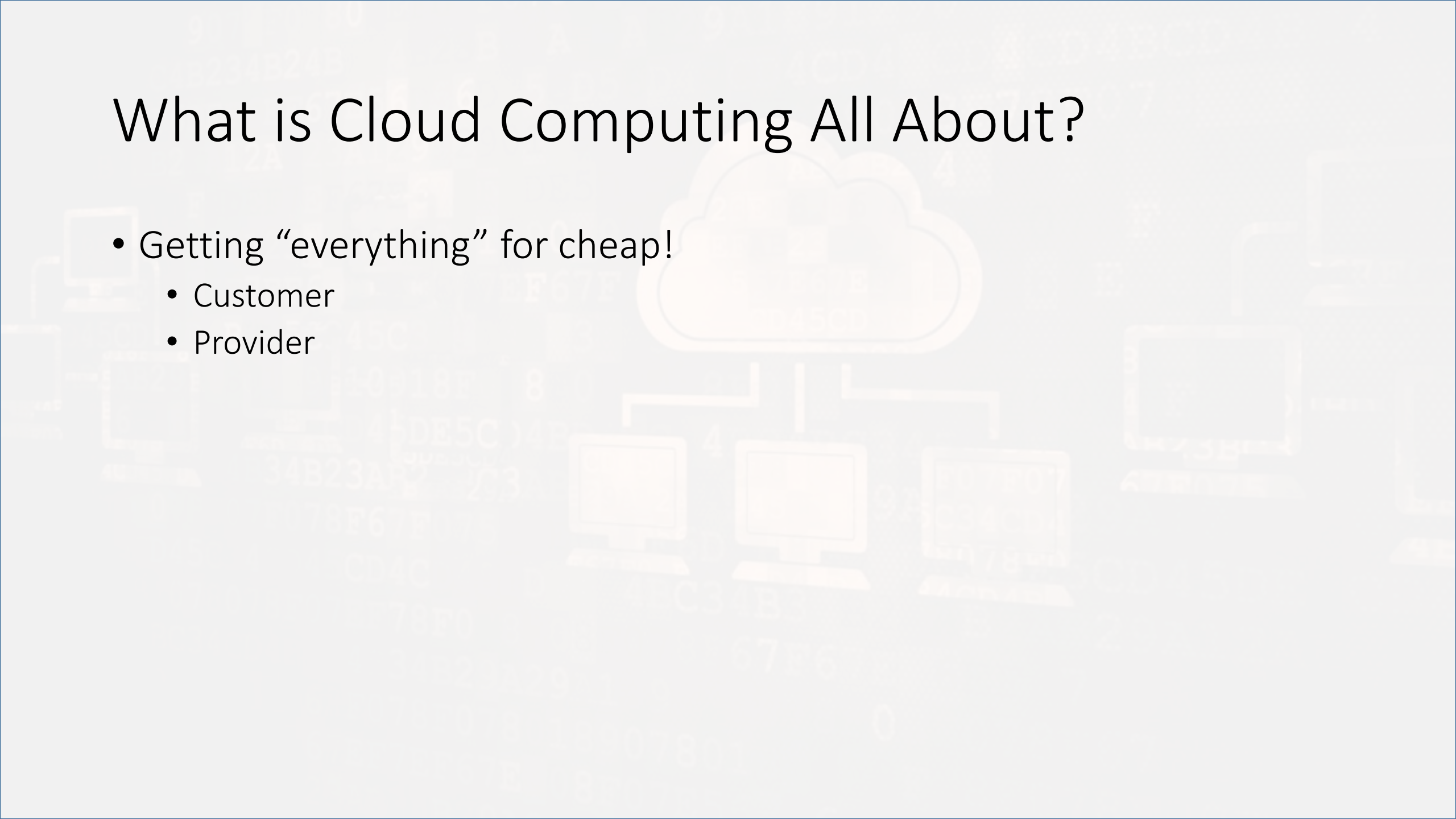
IT: Information Technology

What is Cloud Computing All About?

- Sing along…

What is Cloud Computing All About?

- Getting “everything” for cheap!
  - Customer
  - Provider

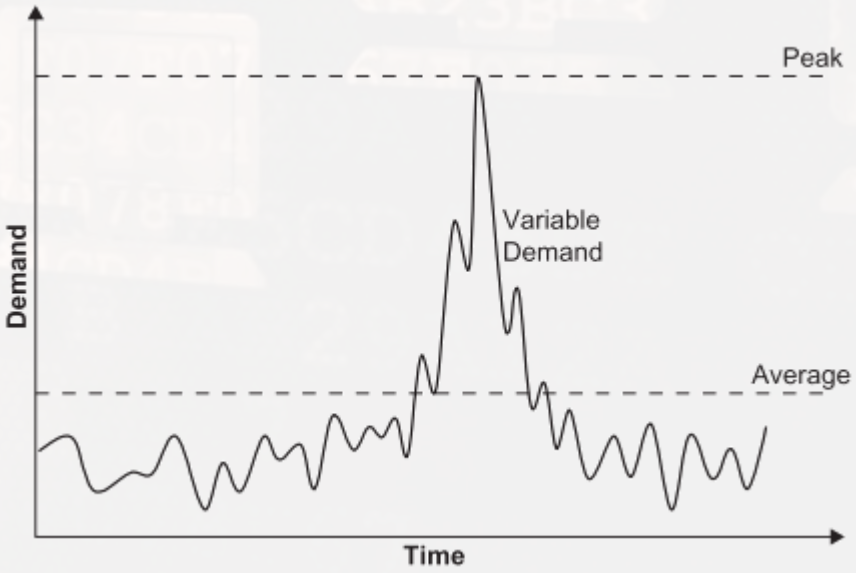


Economy as a Driving Force: Customer Side

Weinman, “Cloudonomics”

- Elements of the game:
  - Demand for resources (time $t \in [0, T]$): $D(t)$
  - Peak demand: P
  - Average demand: A

  - Bursty demands:
    - Breaking news
    - Clearance
    - new products / hypes
    - holiday shopping
    - new movies
    - Release deadline in your startup
    - …

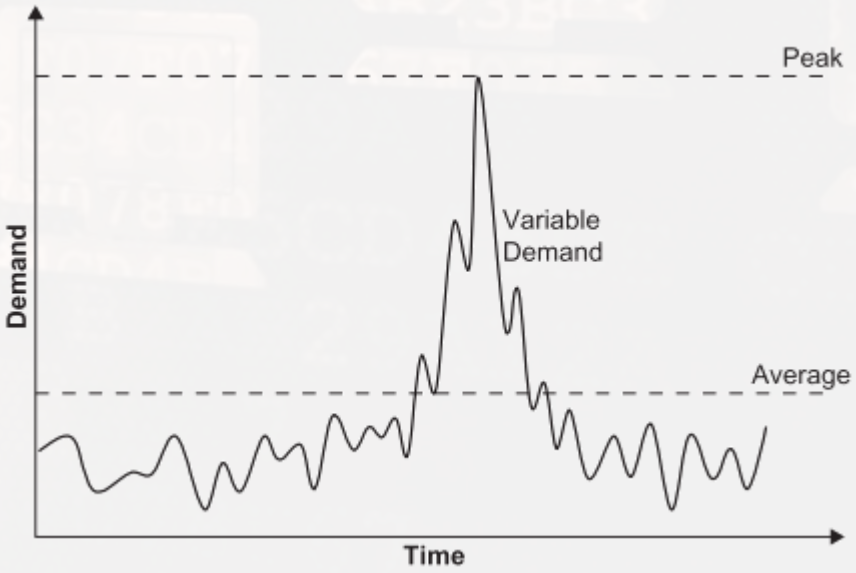


Economy as a Driving Force: Customer Side

Weinman, “Cloudonomics”

- Elements of the game:
  - Demand for resources (time $t \in [0, T]$): $D(t)$
  - Peak demand: P
  - Average demand: A

  - Baseline owning cost/resource unit/time unit: B
    - Like “daily” cost of owning a car: \$15K/5y ~ 10\$/day
  - Cloud cost/resource unit/time unit: C
    - Like daily cost of renting a car. Typically $B < C$
  - Utility premium: $U = C/B$
    - Ratio between daily renting cost to owning cost

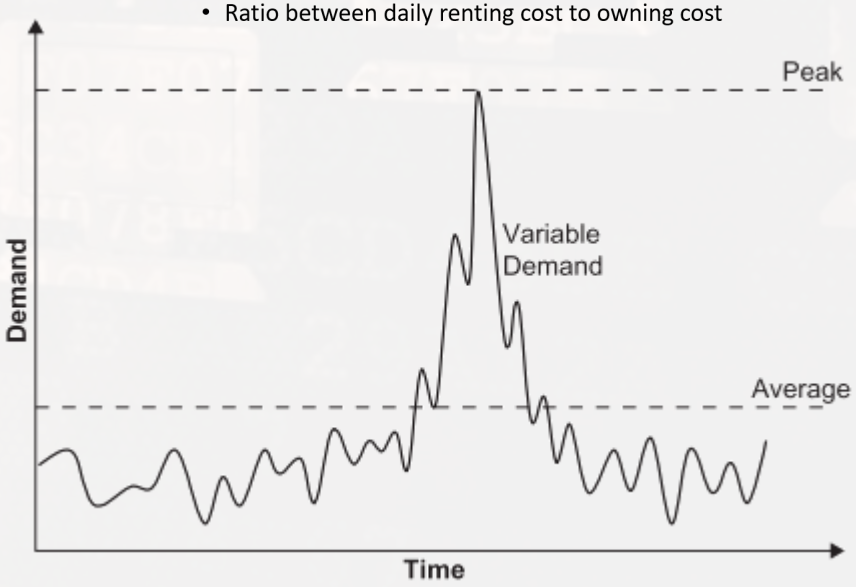


Economy as a Driving Force: Customer Side

Weinman, "Cloudonomics"

- Goal:
  - calculate the total costs of owning (B_T) vs. cloud (C_T)
- Total owning cost: must support peak at all times
  - $B_T = P \cdot B \cdot T$
- Total cloud cost: dynamically changes
  - $C_T = \int_0^T C \cdot D(t)\, dt = U \cdot B \cdot A \cdot T$
- When is it cheaper to use the cloud?
  - $C_T < B_T \Leftrightarrow U < \frac{P}{A}$
- "Know thy workload…"

- Elements of the game:
  - Demand for resources (time $t \in [0, T]$): $D(t)$
  - Peak demand: P
  - Average demand: A
  - Baseline owning cost/resource unit/time unit: B
    - Like "daily" cost of owning a car: \$15K/5y ~ 10\$/day
  - Cloud cost/resource unit/time unit: C
    - Like daily cost of renting a car. Typically $B < C$
  - Utility premium: $U = C/B$
    - Ratio between daily renting cost to owning cost



Economy as a Driving Force: Customer Side

Weinman, “Cloudonomics”

- We’re all customers:
  - Which cable/cellular plan best suits me?
  - Should I get this appliance/gadget/thingy?
    - Dishwasher, new video game, bigger & expensive car, …
- Know thy “workload”!!

Economy as a Driving Force: Provider Side

Weinman, “Cloudonomics”

- Assume some demand distribution D
  - Mean μ_D
  - Standard deviation σ_D
- Coefficient of variation (CV) / relative standard deviation (RSD):

$$C_v(D) = \frac{\sigma_D}{|\mu_D|}$$

  - Measure of smoothness: small is smooth!
    - Large $|\mu_D|$, or small σ_D
    - Caveat: be careful with (very) small $|\mu_D|$…
  - Dimensionless: just a ratio
    - Sometimes measured in %, although might be larger than 100%...
    - Useful for comparing distributions of different types/measures
      - E.g., Celsius vs. Farenheit

Economy as a Driving Force: Provider Side

Weinman, “Cloudonomics”

- Assume some demand distribution D
  - Mean μ_D
  - Standard deviation σ_D
- Coefficient of variation (CV) / relative standard deviation (RSD):

$$C_v(D) = \frac{\sigma_D}{|\mu_D|}$$

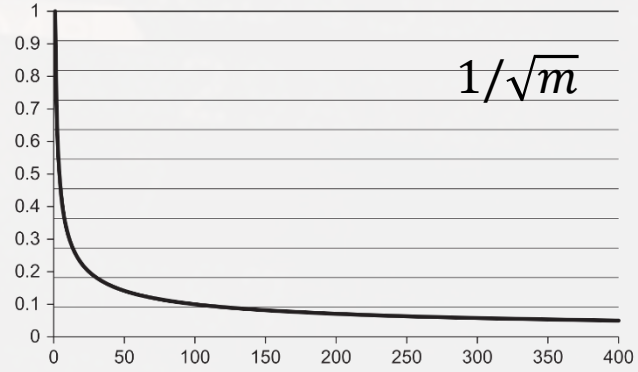
- Examples:

							σ	μ	C_v
D_1	10	10	10	10	10	10	0	10	0
D_2	5	15	5	15	5	15	5.48	10	0.55
D_3	5	15	0	20	-5	25	11.83	10	1.18
D_4	-5	30	-5	40	20	15	18.28	15.83	1.15
D_5	280	290	275	280	300	295	9.83	286.67	0.0342

Economy as a Driving Force: Provider Side

Weinman, “Cloudonomics”

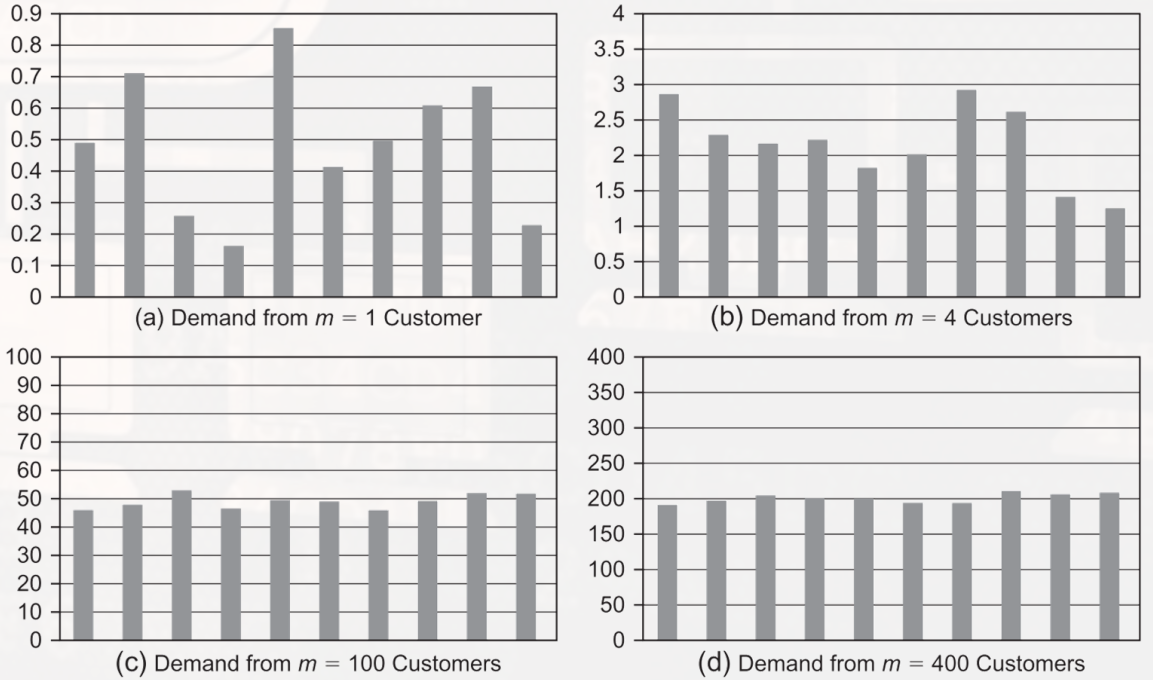
- Provider has fixed amount of resources (provisioned for some target)
  - Servicing highly variable demand ⇒ lower resource utilization / high SLA violation rate
  - If we were able to serve “smoother” demands ⇒ better performance (predictability)
- Assume m independent demands, $D_1, \ldots, D_m$
  - Each with mean μ and standard deviation σ
  - Coefficient of variation: $C_v = \sigma / |\mu|$
- Consider the aggregated workload $\sum_{i=1}^{m} D_i$
  - Mean: $m \cdot \mu$
  - Variance (independent): $m \cdot \sigma^2$
  - Standard deviation: $\sqrt{m} \cdot \sigma$
  - Coefficient of variation: $\frac{1}{\sqrt{m}} C_v$



Economy as a Driving Force: Provider Side

Weinman, “Cloudonomics”

- Statistical multiplexing: makes the overall demand “smoother”
  - For independent demands…
- Best case:
  - Negatively correlated demands
    - $\sum_{i=1}^{m} D_i$ is constant ⇒ aggregate CV is zero
    - E.g., $D_2 = 1 - D_1$
    - Not realistic…
- Worst case:
  - Positively correlated demands
    - All D_i are identical ⇒ same aggregate CV
      - Mean $m \cdot \mu$, standard deviation $m \cdot \sigma$
    - But, statistical multiplexing doesn’t hurt…



What is Cloud Computing, “Officially”?

Mell and Grance, “The NIST Definition of Cloud Computing”, NIST Special Publication 800-145, 2011

"*Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g. networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction*".

What is Cloud Computing, “Officially”?

Mell and Grance, “The NIST Definition of Cloud Computing”, NIST Special Publication 800-145, 2011

"*Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g. networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction*".

- On-demand self-service
  - Customers can provision computing capabilities, as needed, automatically

What is Cloud Computing, “Officially”?

Mell and Grance, “The NIST Definition of Cloud Computing”, NIST Special Publication 800-145, 2011

"*Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g. networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction*".

- Broad network access
  - Capabilities available over the network, accessed through standard mechanisms by heterogeneous client platforms
    - Clients may range from servers to smartphones

What is Cloud Computing, “Officially”?

Mell and Grance, “The NIST Definition of Cloud Computing”, NIST Special Publication 800-145, 2011

"*Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g. networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction*".

- Resource pooling
  - Computing resources pooled to serve multiple consumers, virtual resources dynamically (re)assigned
    - Compute, network, storage
  - Customers unaware of actual physical resources to which they are assigned
    - May sometimes specify general requirements (region, country, datacenter)

What is Cloud Computing, “Officially”?

Mell and Grance, “The NIST Definition of Cloud Computing”, NIST Special Publication 800-145, 2011

"*Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g. networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction*".

- Rapid elasticity
  - Capabilities elastically provisioned and released, appear unlimited to customer
    - Potentially automatic scale-in / scale-out

What is Cloud Computing, “Officially”?

Mell and Grance, “The NIST Definition of Cloud Computing”, NIST Special Publication 800-145, 2011

"*Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g. networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction*".

- Measured service
  - Resource usage monitored, controlled, and reported (both provider and consumer)

Service Models

Mell and Grance, “The NIST Definition of Cloud Computing”, NIST Special Publication 800-145, 2011

- Software as a Service (SaaS)
- Platform as a Service (PaaS)
- Infrastructure as a Service (IaaS)

- More generally: Anything as a Service (XaaS)
  - Security (S), Monitoring (M), Business Process (BP), Artifical Intelligence (AI), …
  - Even Function-as-a-Service (FaaS)…
  - Specific “product”, pay for what you use

Service Models

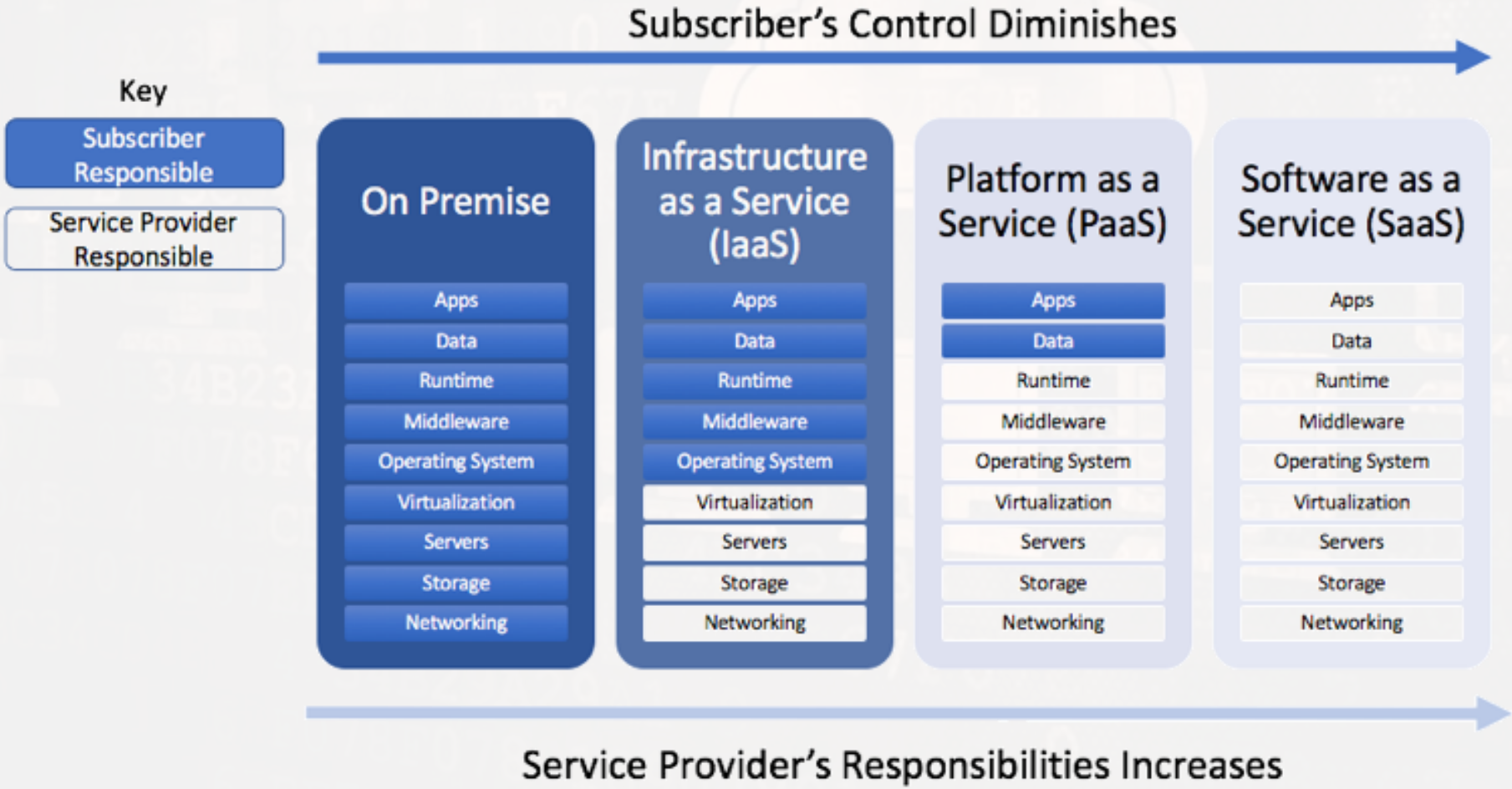


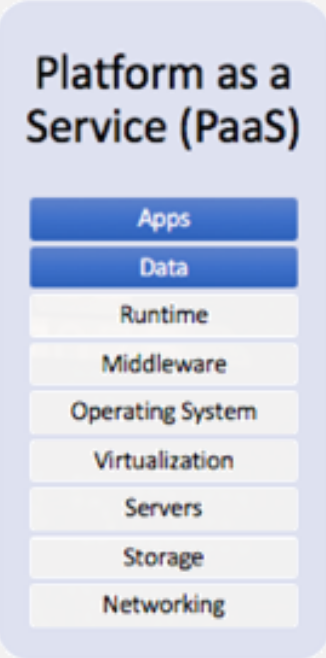
Image: Chris Lazari

SaaS

- Provider is in charge of everything
- Customer can just use the software/application
- Predominantly web-services
  - OS-independent, access from anywhere, sharing
  - Low maintenance: no worries of outage, storage, backups
- Examples:
  - Google mail (e.g, in BGU)
  - WiX
  - Dropbox
  - …

PaaS

- Customer can focus only on its application (and data)
  - Specifies what underlying platforms it requires
    - Libraries, OS, programming language, …
- Provider handles everything else:
  - Server patching, Load balancing, scaling, DBs, …
  - Support various development environments
- Features:
  - Better resource management for Provider
    - Multi-tenancy / co-location
  - Risk of vendor lock-in (e.g., due to proprietary APIs)
- Examples:
  - Google App Engine



IaaS

- Customer is in charge of everything but the HW
  - Entire SW stack
  - Networking configuration, load balancing
- Features
  - Better separation across tenants
    - Separate VM for each customer
  - (Much?) more overhead for managing workloads
- Examples:
  - AWS EC2 (Elastic Compute Cloud) / S3 (Simple Storage Service)
  - Microsoft Azure
  - Google Compute Engine

(Some Common) Deployment Models

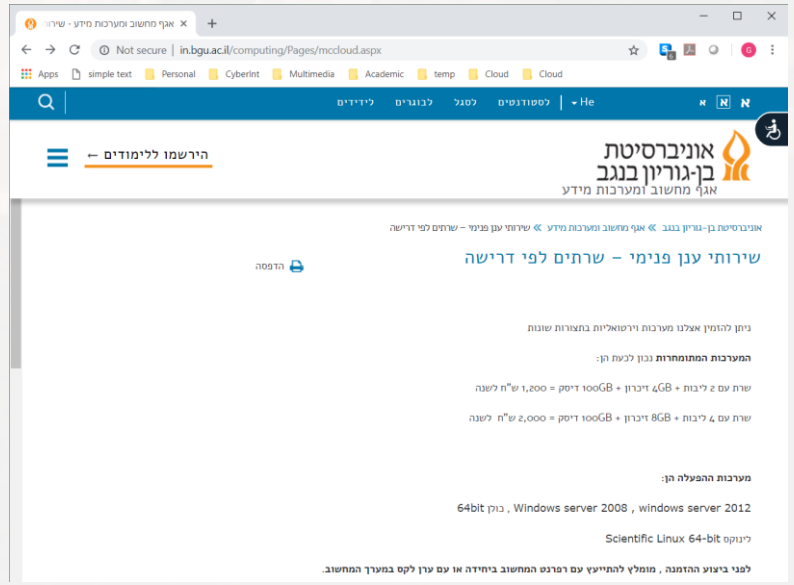
Mell and Grance, “The NIST Definition of Cloud Computing”, NIST Special Publication 800-145, 2011

- Private cloud
  - Owned, managed, and operated for usage by a single organization
  - Community cloud
    - “Private” to several organizations, usually with shared concerns
      - E.g., mission, security, policy, and compliance considerations
- Public cloud
  - Cloud infrastructure is provisioned for open use by the general public
- Hybrid cloud
  - Composition of two or more distinct cloud infrastructures (private/public)
  - Data and application portability:
    - Possible using standardized or proprietary technologies

Private Cloud

Badger et al., “Cloud Computing Synopsis and Recommendations”, NIST Special Publication 800-146, 2012

- Organization “owns” all HW/SW
  - Deployment, maintenance, scaling
  - Sometimes managed by a provider
    - Still, all HW is dedicated and paid-for upfront
- Benefits
  - Control, security, abiding to regulation (if required)
- Downside: cost…
- Private cloud ≅ Private data center
- Examples:
  - Large corporations
  - BGU’s private cloud
  - Course’s private cloud…



Public Cloud

Badger et al., “Cloud Computing Synopsis and Recommendations”, NIST Special Publication 800-146, 2012

- Service-based pay-per-use
  - Control, security, load balancing, maintenance, …: depends on service
  - Internet-based access
- Benefits
  - Dynamic demands, flexible scaling
  - Cost!!
- Examples
  - Amazon AWS
  - Microsoft Azure
  - Google Cloud

Outline

- Syllabus and Course Administration
- What is Cloud Computing
- What is Virtualization
- Cloud Miscellany

Virtualization: Cloud Computing Workhorse

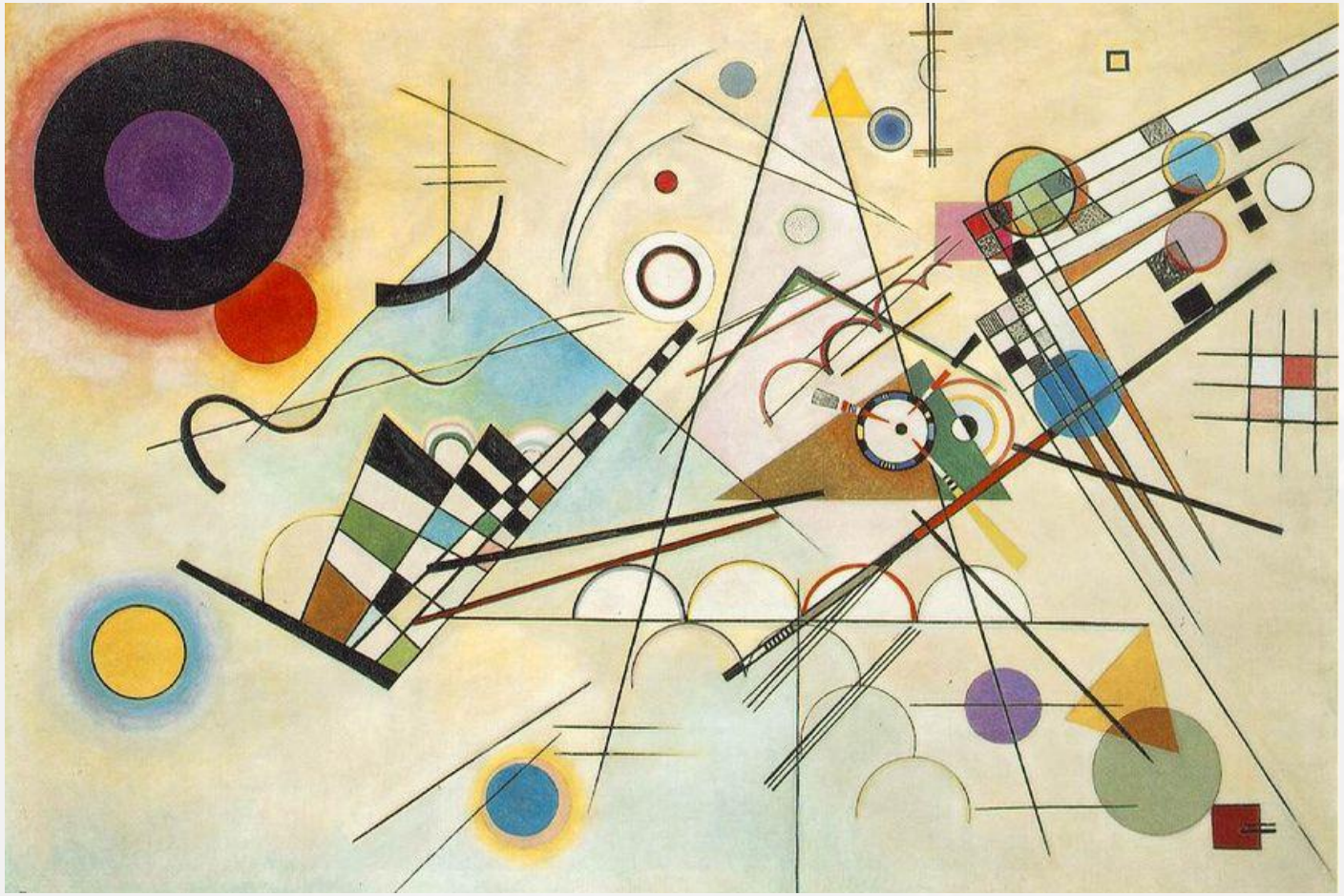
- Raison d'être:
  - Distributed computing without dependency on physical resources
- Goal:
  - Being an excellent fraud
    - Make you think you're seamlessly using the physical resources directly, when actually, you're not…
- What are we virtualizing?
  - Compute:
    - Physical servers, with multiple capabilities (cores, memory, GPUs, etc.)
  - Storage:
    - Physical disks, disk access
  - Network
    - Bandwidth, connectivity, switches, routers
    - Network functions and services

(Some) Virtualization Terminology

- Host
  - Physical host, with some specific physical HW configuration
  - May run its own SW (with/without an underlying host OS) to provide virtualization
- Guest
  - A user wishing to perform computation on a machine
- Virtual Machine (VM)
  - A complete guest environment running on a host
    - VM: has its own guest OS
- Hypervisor / VMM (Virtual Machine Monitor)
  - SW module responsible for running VMs (and their guest OS) on the physical HW / host OS
- Virtual CPU (vCPU)
  - VM CPU state, maintained by the VMM
- Tenant
  - A user wishing to perform computation in the system
  - Potentially on some “cluster” of computation resources

(will later see “containers”, which are somewhat different)

Virtualization: The Art of Abstraction



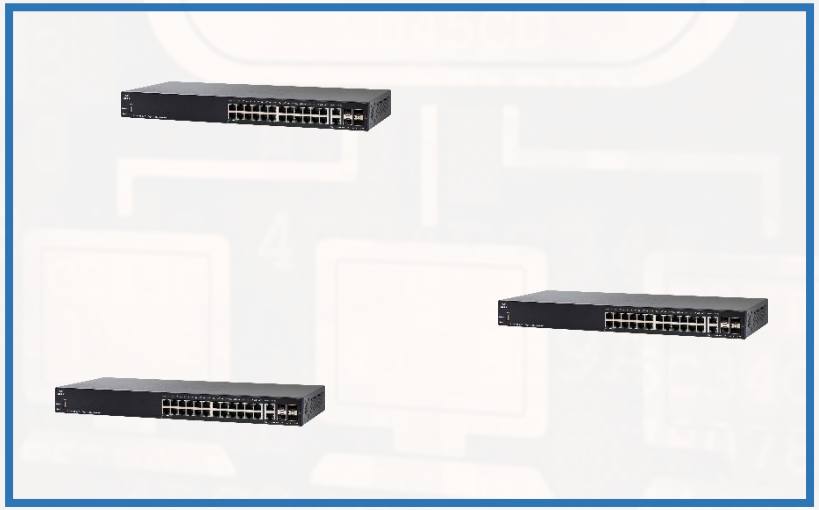
Wassily Kandinsky, Composition 8, 1923

Virtualization: The Art of Abstraction (Illustration)

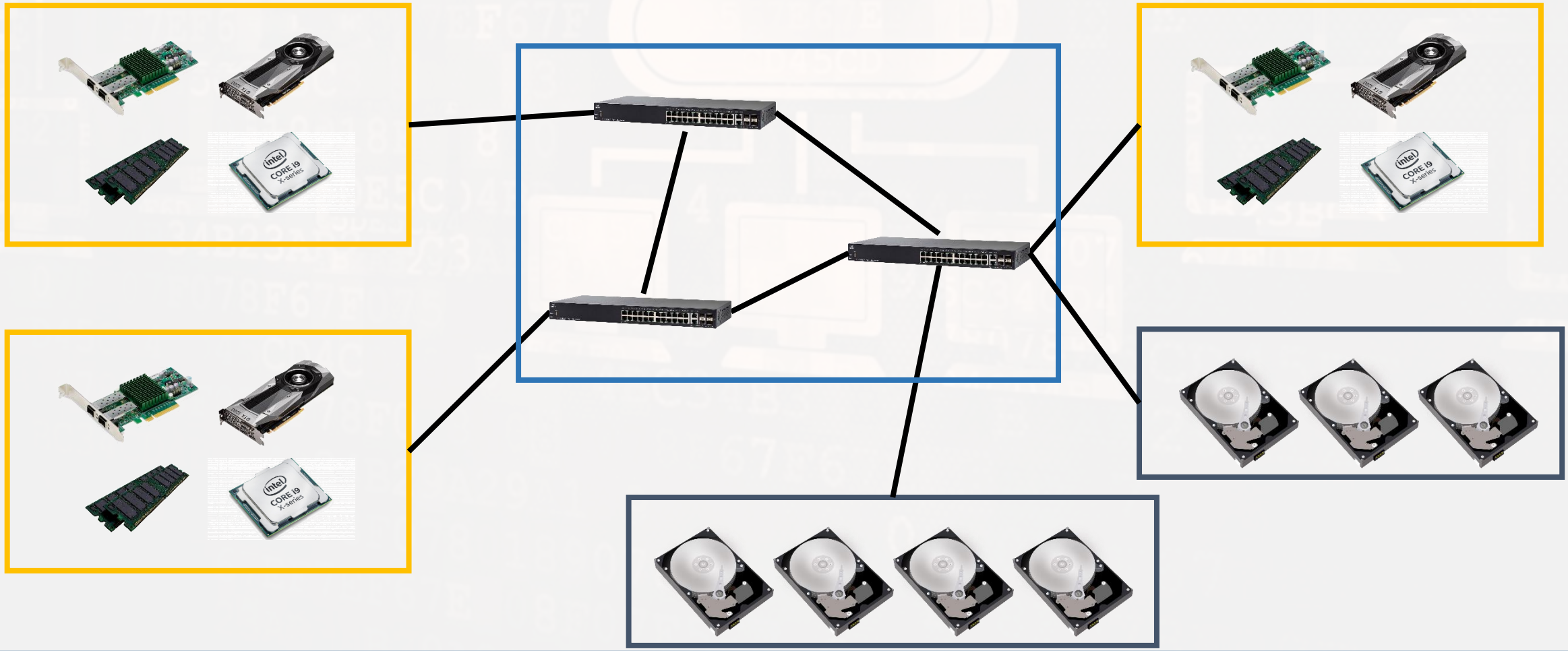
- HW components
  - CPU
  - Memory
  - Disk
  - NICs
  - Networking HW (switch, router)
  - GPU
  - …

Virtualization: The Art of Abstraction (Example)

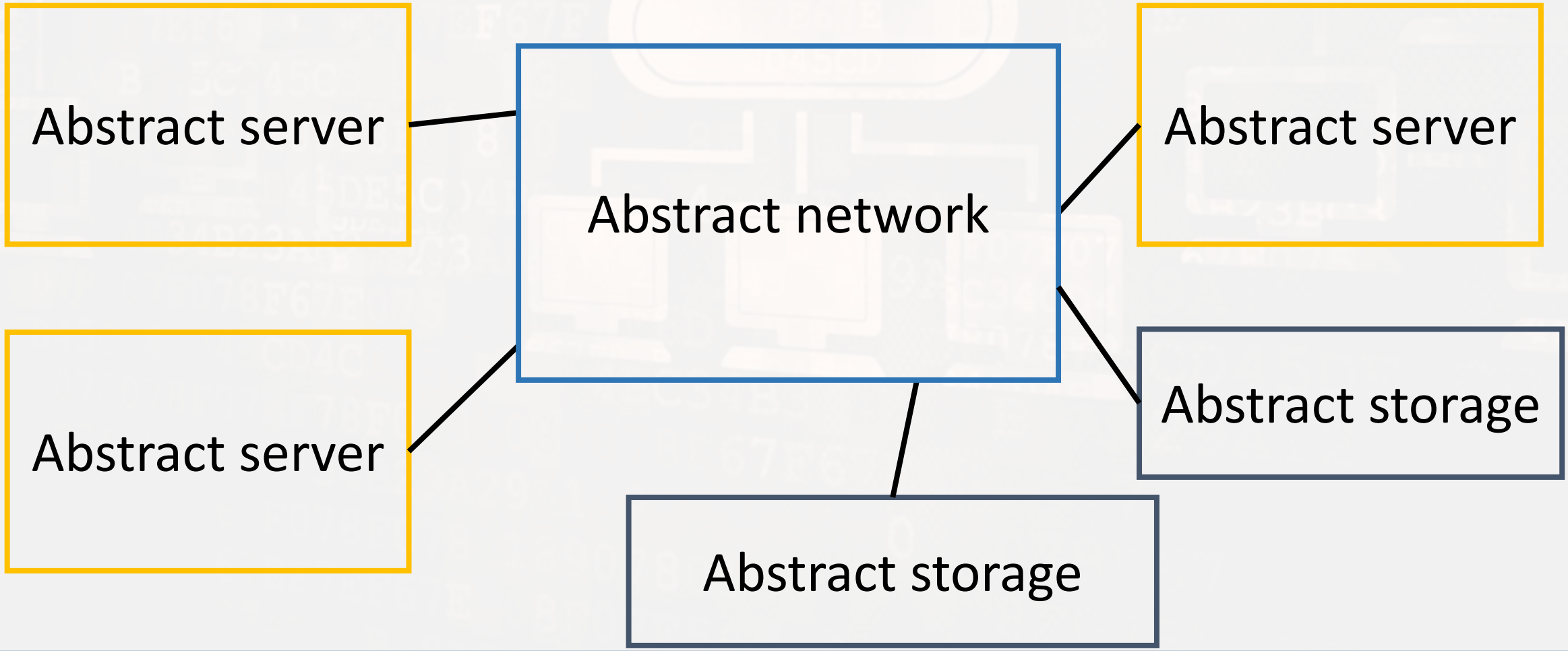
Virtualization: The Art of Abstraction (Example)



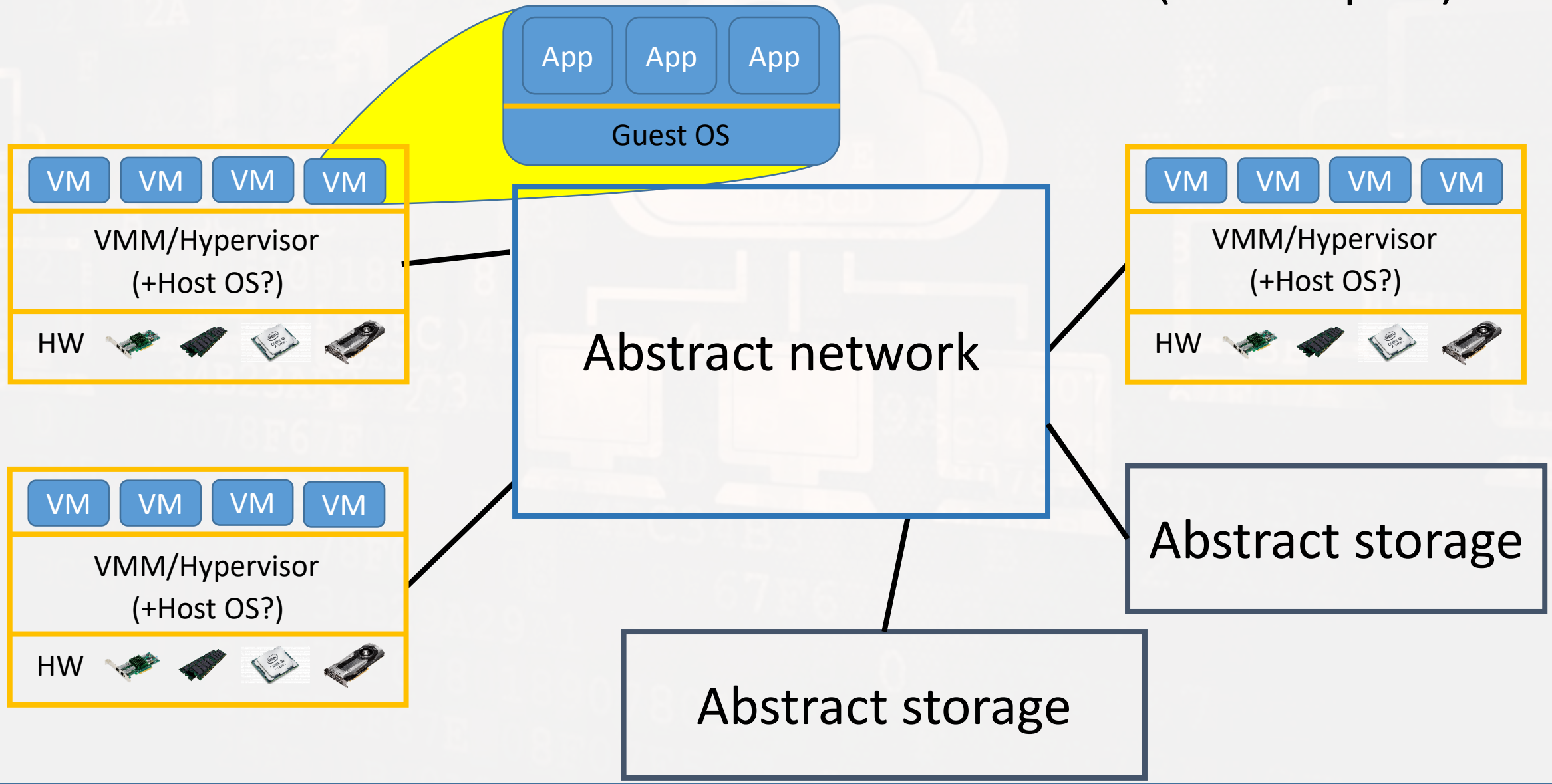
Virtualization: The Art of Abstraction (Example)



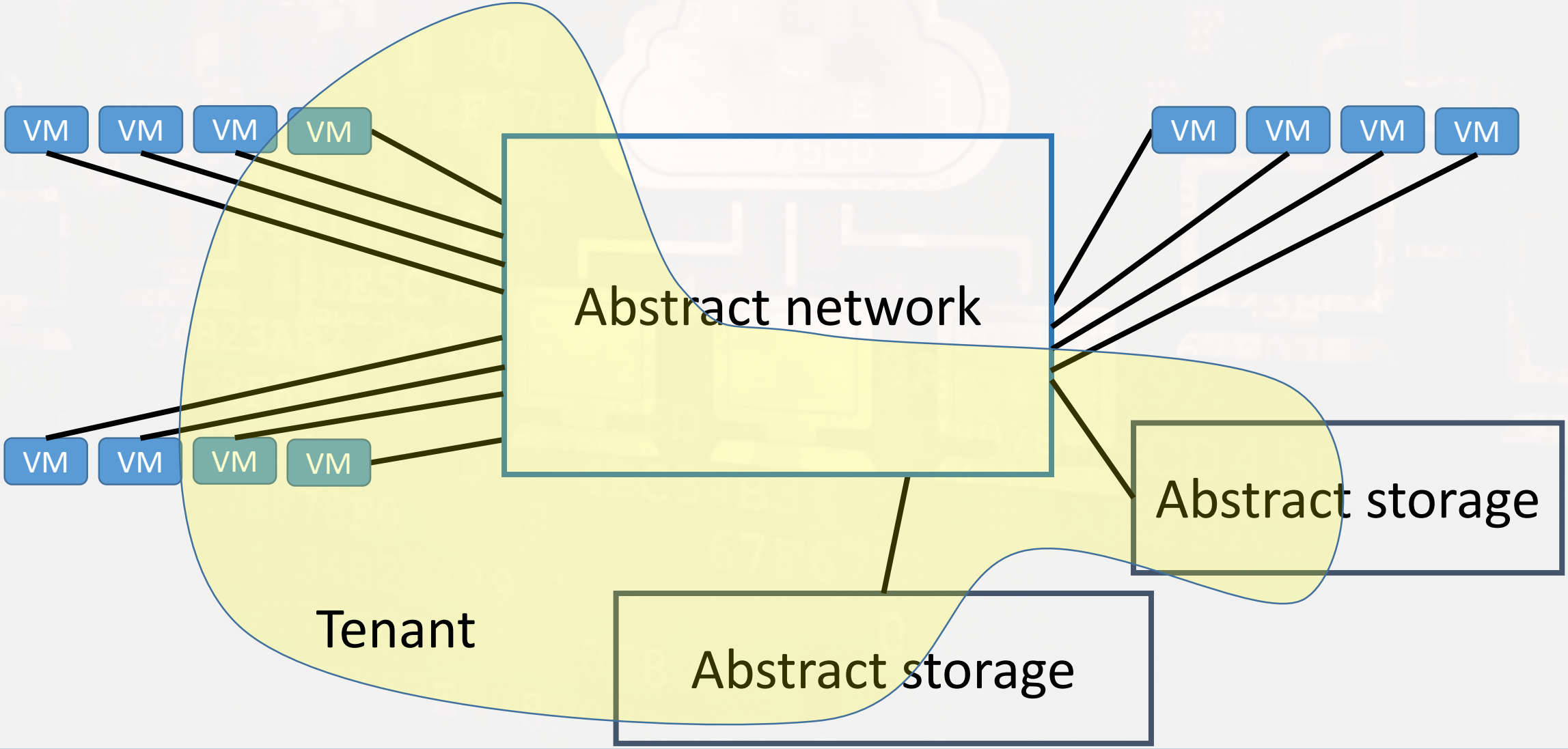
Virtualization: The Art of Abstraction (Example)



Virtualization: The Art of Abstraction (Example)



Virtualization: The Art of Abstraction (Example)



Virtualization: “Formal” Definition

- Popek & Goldberg (1974)
- Efficient virtualization should satisfy:
  - Equivalence: VMM ≡ HW for any VM
  - Resource control: VMM controls all virtualized resources
  - Efficiency: major portion of VM instructions are executed natively
    - I.e., without VMM intervention
- Is virtualization possible?
  - Depends on Instruction Set Architecture (ISA)
  - Various techniques for satisfying the requirements (stay tuned…)
- Efficiency: different techniques have different pros/cons
  - No silver bullet

Virtualization for Development and Security

- Sandboxing
- Development
  - Code development within a VM
    - OS crashes only require restarting VM
  - Extremely useful when writing/debugging OS-code
    - E.g., using emulators such as BOCHS
    - Allows “re-playing” entire activity from within host-OS
- Security
  - Run potentially malicious code within a VM
  - Access potentially malicious servers from within a VM
  - Used extensively in cyber-security

Outline

- Syllabus and Course Administration
- What is Cloud Computing
- What is Virtualization
- Cloud Miscellany

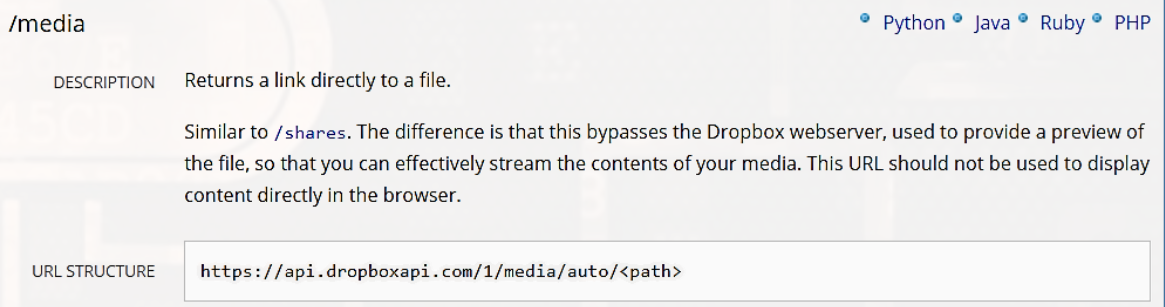
Programming/Computing Paradigm

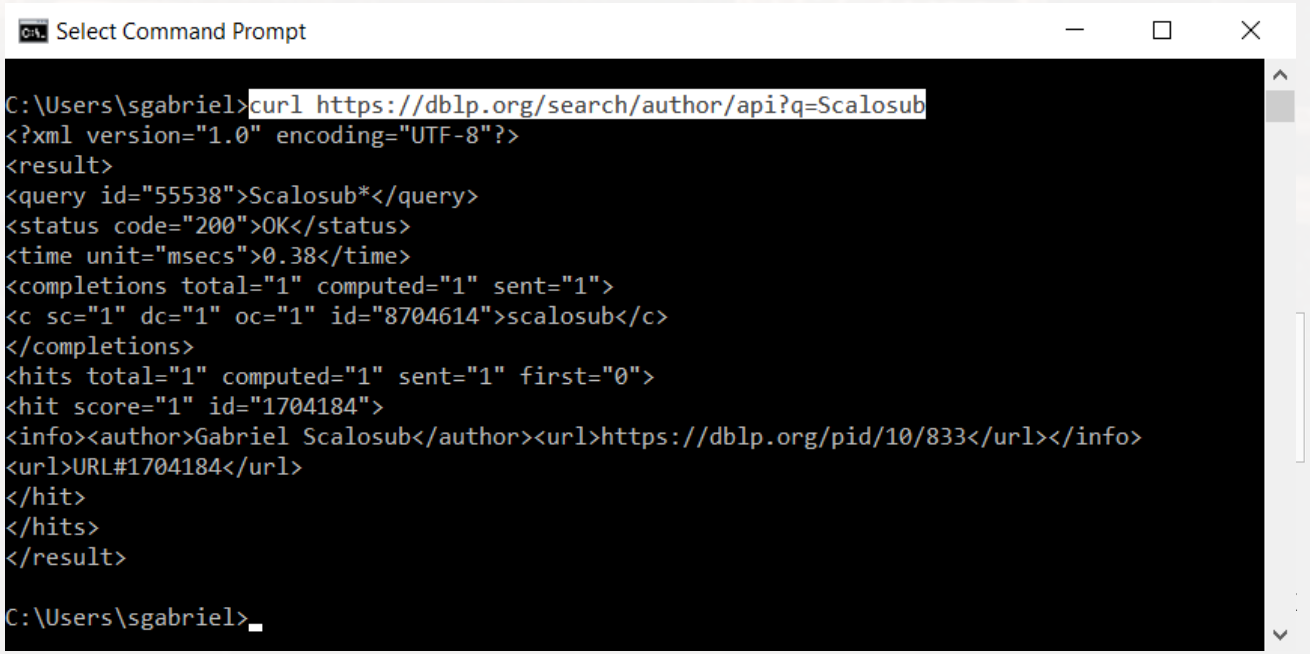
Badger et al., “Cloud Computing Synopsis and Recommendations”, NIST Special Publication 800-146, 2012

- Client-server
  - Clients send messages to servers over a network
  - Servers perform work/computation
    - Possibly in a distributed manner
  - Clients receive response from servers
- Cloud
  - Manages pool of computing resources
    - For efficiency, fault-tolerance/redundancy
    - Assigns and migrates workloads
  - Network dependency

RESTful API

- REST: REpresentational State Transfer
- API: Application Program Interface
- RESTful API:
  - HTTP-based interface
    - GET, POST, PUT, DELETE
    - To be sent to the server
  - Includes parameters, and returned output





Some Applications (AKA, buzzwords)

- Big Data: V^3 -> V
  - Volume (large), velocity (increases fast), variety (very diverse) -> Value!!
  - Search engines, distributed databases, …
- Machine learning
  - Parallel ML tasks, handling huge datasets
  - Deep learning with neural networks
- 5G: Future cellular networks
  - C-RAN (Cloud / Centralized Radio Access Network)
  - Centralized Baseband Units (BBUs) pool, Remote Radio Heads/Units (RRH/RRU)
- Edge/Fog computing
- Internet-of-Things (IoT)
- Many more…

Cloud Security & Privacy

- Main dilemmas:
  - Multi-tenancy vs. isolation
    - Secure against co-located VMs
    - Affected by type/level of virtualization
    - Security-performance tradeoffs
  - Customer-Provider
    - Security and privacy issues: hosting data “outside the perimeter”
    - Affected by deployment model (private vs. public), provider security practices
- Concerns:
  - Data breach, VM escape, side-channel attacks, …

HPC (High-Performance Computing)

- Supercomputers
  - Hard computational tasks
    - “big data”
  - Medical research
    - Genomics, Imaging
    - Personalized medicine
  - Physics
    - Nuclear, Quantum
  - Engineering
    - Large scale simulation
    - E.g., airplane design
  - Finance
    - Insurance

Robert Redford & Ben Kingsley sitting on Cray Supercomputer
“Sneakers” (1992)

Cloud vs. HPC

- 200 people to clean festival grounds

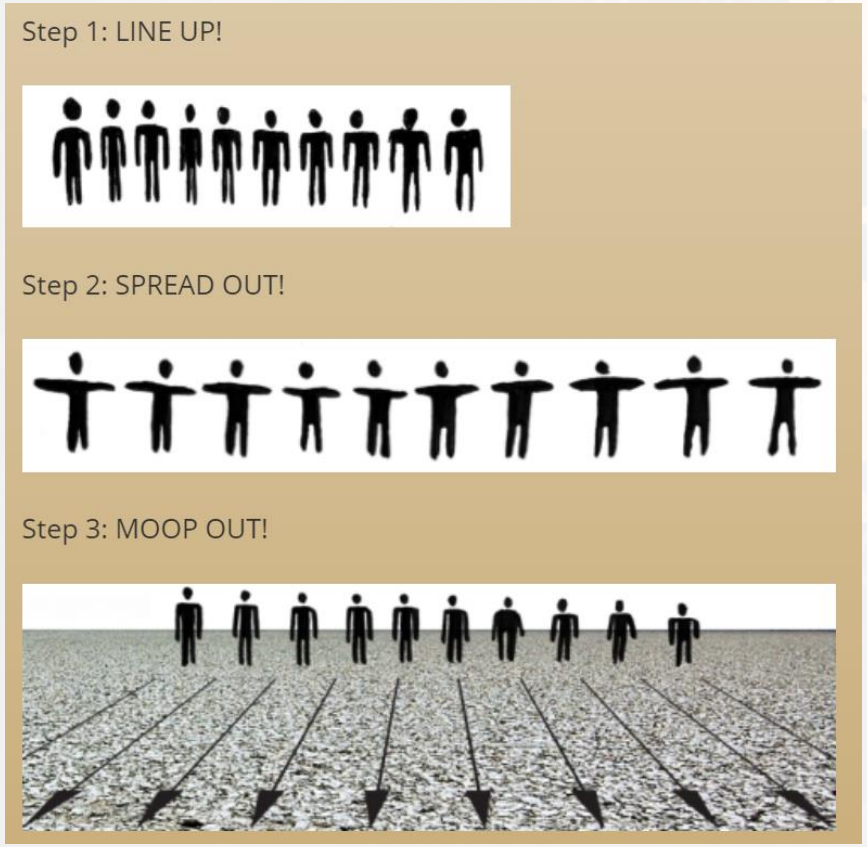


Image: https://burningman.org/

- Easily scalable
  - Very little “coordination”
  - Each one operating alone, locally
    - Minimal “interaction”
    - Just don’t step on toes…
  - AKA Embarrassingly/pleasingly Parallel
    - Web server
    - Brute-force cryptography
    - Hyperparameters grid search
    - …
- But…

Cloud vs. HPC

- 200 people flashmob

Cathay Pacific Flashmob @ HKIA 2013

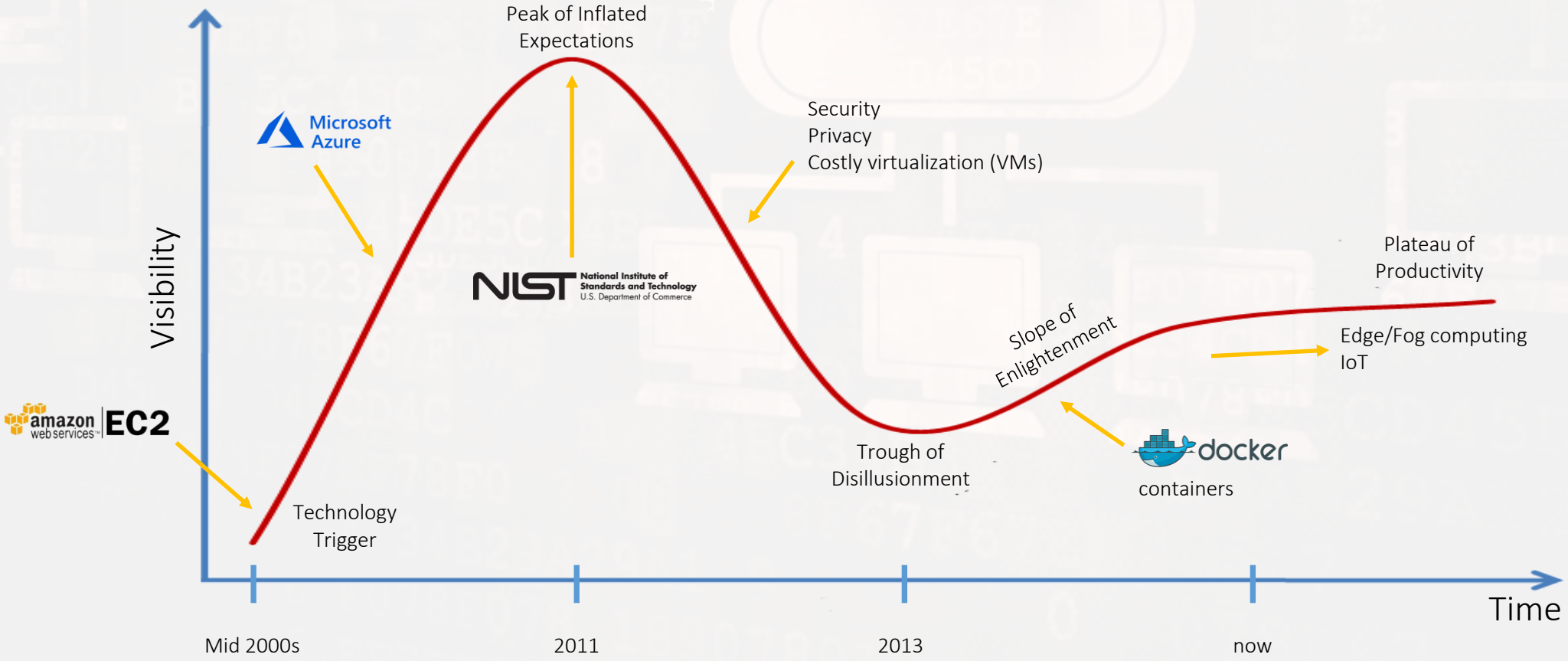
- Harder to scale
  - Lots of coordination/practice
    - Not “off-the-shelf”…
    - Constant communication
  - Carefully share resources
  - Precision/timing is everything

Cloud vs. HPC

- High-performance computing
  - Computer clusters performing single computing task in parallel
  - Joint datasets
  - Costly workload deployment
    - Cluster setup, cost independent of usage
  - “Virtual” supercomputer
    - Usually (very) high-performance servers
    - High-speed data connections
      - E.g., Infiniband
      - Bypassing OS kernel
  - Vertical scalability: Single (big) application runs on various nodes
- Cloud
  - Support multiple tenants simultaneously
  - “Independent” tasks/applications
  - Cost-effective deployment
    - Pay-per-usage, fast setup
  - “Simple” computation platform
    - Doesn’t necessarily require high-end servers
    - Standard off-the-shelf HW/SW
    - Virtually infinite resources
  - Horizontal scalability: Multiple applications (instances) run on nodes

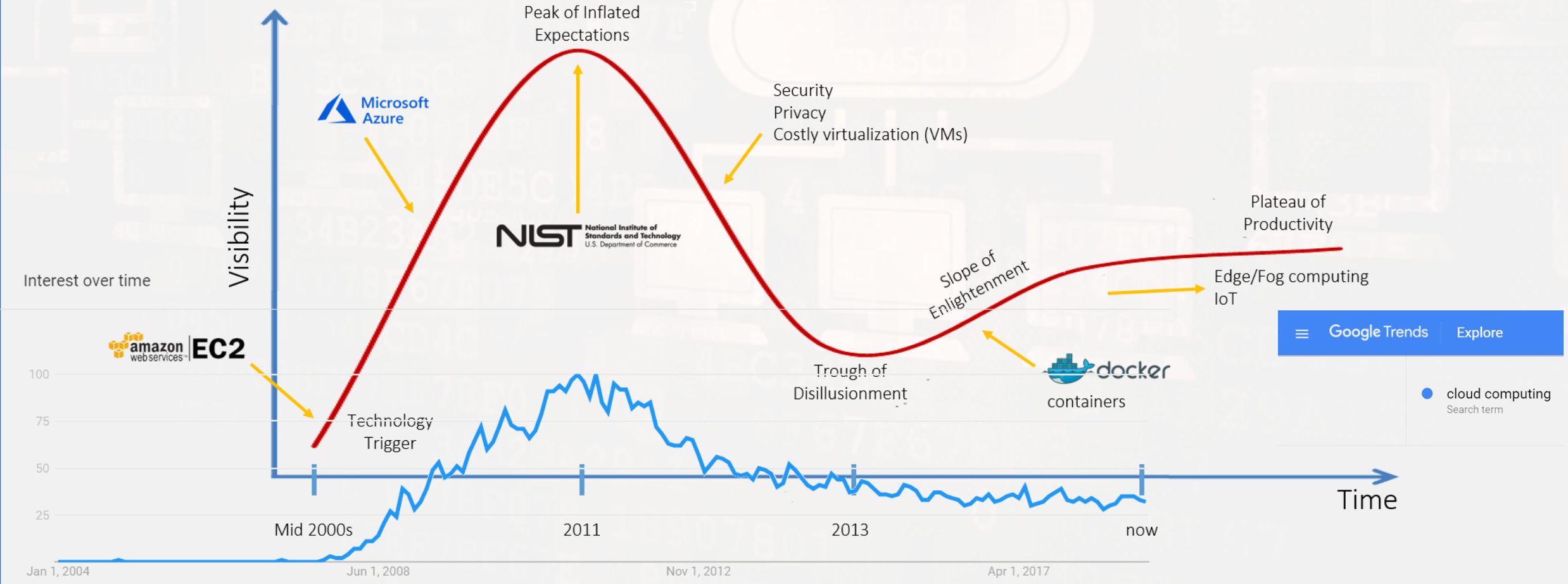
Hype Cycle (for Cloud)

Based on “Gartner Hype Cycle”



Hype Cycle (for Cloud)

Based on “Gartner Hype Cycle”



(Partial) Bibliography

- Mell & Grance, “The NIST Definition of Cloud Computing”, NIST Special Publication 800-145 (2011)
- Badger et al., “Cloud Computing Synopsis and Recommendations”, NIST Special Publication 800-146 (2012)
- Weinman, “Cloudonomics: The Business Value of Cloud Computing”, Wiley (2012), Chapters 11, 15
- Barroso et al., "The Datacenter as a Computer: An Introduction to the Design of Warehouse-Scale Machines", M&C Publishers (2013)
- Doeppner, “Operating Systems In Depth: Design and Programming”, Wiley (2011)
- Tanenbaum & Bos, “Modern Operating Systems”, 4th ed., Pearson (2014)