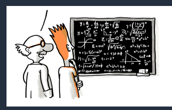
DSC516: Cloud Computing

Part I: Basic Concepts and Models

Module 2: Cloud Computing Definitions and Models

In previous lectures



- Discussed the key technological and economic developments that led to Cloud Computing Infrastructures.
- Explored and explained the concepts and role of:
  - Moore's Law, Mainframes, PCs and Client-Server
  - Cluster Computing, Web Computing, Internet-scale Services, Exponential Phenomena, Network Effects
  - General Purpose Technologies and Public Utilities
  - Grid Computing, Utility Computing, Software-as-a-Service
- Discussed and explained some basic concepts of distributed computing systems:
  - Abstraction, Architecture, System Architecture
  - Resources, Physical and Logical, Process
  - Distributed Computing Models: Client-Server, REV, COD, MA
  - Middleware services and categories
  - End-to-end arguments in system design and functional decomposition, performance tradeoffs, applying e-2-e arguments in various application scenarios

Lecture 3

Cloud Computing: Introduction, Definitions, Taxonomy

Readings

- Barroso, L. A., & Holzle, U. (2015). **The Datacenter as a Computer. An Introduction to the Design of Warehouse-Scale Machines.** In Synthesis Lectures on Computer Architecture (Vol. 2, Issue 1). Morgan & Claypool Publishers. Chapters 1-3.
- M. Armbrust et al., **"A view of cloud computing,"** Communications of the ACM, vol. 53, no. 4, p. 50, 2010. https://doi.org/10.1145/1721654.1721672
- Jim Gray, **"Distributed Computing Economics"**, J Microsoft-TR-2003-24, 2003.
- O. Agmon Ben-Yehuda, M. Ben-Yehuda, A. Schuster, and D. Tsafrir, **"The rise of RaaS,"** Commun. ACM, vol. 57, no. 7, pp. 76–84, Jul. 2014.

Cloud Computing: Introduction, Definitions, Taxonomy

Key Features

Cloud Computing

- Refers to both the applications delivered as services over the Internet and the hardware and systems software in the data centers that provide those services.
- Related terms:
  - Software-as-a-Service: applications (services) delivered over the Internet, **on-demand**.
  - Grid Computing: federated data centers, and associated protocols to offer shared computation and storage over long distances (HPC-community driven).

The "Cloud"

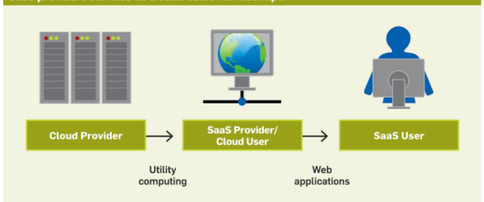
Data center hardware and **software**

- Public cloud: a cloud that is made available in a pay-as-you-go manner to the general public.
  - Utility Computing: the service being sold by a public cloud.
- Private clouds: internal data centers, not made available to the general public, but large enough to benefit from the advantages of cloud computing

Cloud Computing: a Description

- Cloud computing: SaaS + Utility computing
- Does **not** include small or medium- sized data centers, even if these rely on virtualization for management.

Figure 1. Users and providers of cloud computing. We focus on cloud computing's effects on cloud providers and SaaS providers/cloud users. The top level can be recursive, in that SaaS providers can also be a SaaS users via mashups.



*M. Armbrust et al., "**A view of cloud computing,**" Communications of the ACM, vol. 53, no. 4, p. 50, 2010*

The New aspects

From a hardware provisioning and pricing point of view:

1. Appearance of infinite computing resources available on demand, quickly enough to follow load surges, eliminating the need to plan far ahead for provisioning.
2. Elimination of up-front commitment by cloud users.
3. Ability to pay for use of computing resources on a short-term basis as needed and re-lease them as needed.

- Prior failures of utility computing: one or two of these characteristics not met.

Essential Cloud Characteristics

- Ubiquitous, convenient, on-demand network access
- Minimal interaction with the service provider
- Minimal management effort
- Location-transparent, shared pool of configurable resources
- Rapid provisioning and release (elasticity)
- Measured service with pay per use

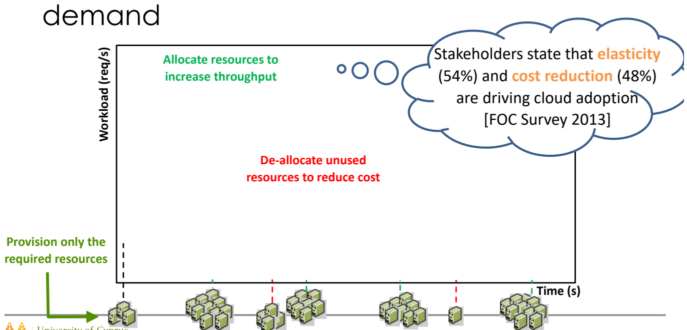
The NIST Definition of Cloud Computing, NIST, 2011

Cloud Computing: Key Features

Elasticity & the Economics of Cloud Provision

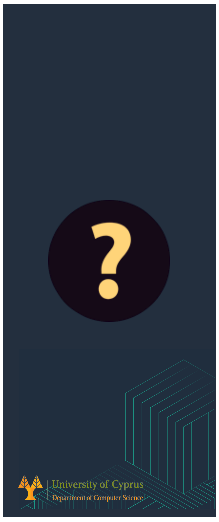
The Concept of Elasticity

- Ability of a system to **expand** or **contract** its dedicated resources to meet the current demand



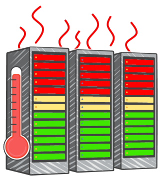
Horizontal vs Vertical Elasticity

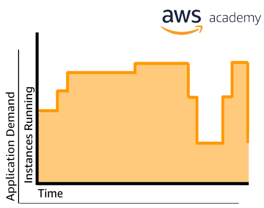
- Horizontal elasticity:
  - Rent/launch Virtual Servers (Machines) for shorter periods of rent, with shorter billing units, and lower overhead in spawning.
  - Reprice computing resources every few secs; charge by the sec.
- Vertical elasticity:
  - Rent and charge for compute, memory, and I/O, in dynamically changing amounts
  - Buy seed VMs with initial amount of resources, supplementing them with additional resources as needed.
- Ideal: rent resources separately with fine resource granularity for short periods (difficult to provide)



WHY IS ELASTICITY IMPORTANT?

The Benefits of Elasticity

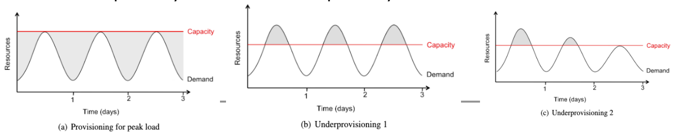




Overestimated server capacity

Underestimated server capacity

Scaling on demand



A note on Server Utilization

- Real world estimates of server utilization in data-centers range from 5% to 20%:
  - for many services the **peak workload** exceeds the **average** by factors of **2 to 10**
- In typical e-commerce services, we see:
  - simple diurnal patterns
  - seasonal or other periodic demand variation
  - some unexpected demand bursts due to external events ("flash crowds")

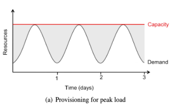
The Economics of Cloud Computing

Converting **capital expenses** (CapEx) to **operating expenses** (OpEx)

- "Pay as you go" (*OpEx*): may be more expensive than buying/depreciating a comparable server (*CapEx*) over the same period but its cost is outweighed by economic benefits of:
  - **Elasticity**: purchasing of resources distributed in time in a non-uniform manner
  - **Transference of risks** of over-provisioning (underutilization) and under-provisioning (saturation) from service operator to cloud vendor: no up-front capital expenses

Overprovisioning

- Assume service has a predictable daily demand where the **peak** requires **500 servers at noon** but the **trough** requires only **100 servers at midnight.**



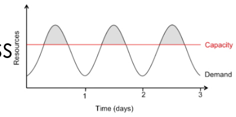
- With average utilization per day to 300 servers, the actual utilization over the whole day is 300 × 24 = 7200 server-hours
- Since we provision to the peak of 500 servers, we pay for 500×24 = 12000 server-hours - 1.7 more than what is needed.
- As long as the pay-as-you-go **cost per server-hour** over 3 years is less than 1.7 times the cost of buying the server, we can save money using utility computing.

The monetary effects of under-provisioning are harder to measure than those of over-provisioning.

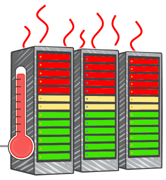
WHY?

Underprovisioning

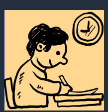
- Underestimate the spike, accidentally turning away excess users.
- Monetary effects harder to measure but potentially equally serious.
- Rejected users:
  - generate zero revenue,
  - may **never come back** due to poor service



(b) Underprovisioning 1



Example



Assumptions of simplified example:

1. Users desert an under-provisioned service until the peak user load equals the data center's usable capacity, at which point users again receive acceptable service.
2. Users fall into two classes:
   - **active users**: use the site regularly
   - **defectors:** abandon the site or are turned away from the site due to poor performance
3. A number of active users equal to the **10%** of those **who receive poor or no service** due to under-provisioning are "permanently lost" opportunities (become defectors).

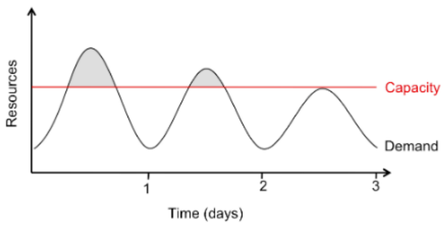
Transference of risk

- The site is initially provisioned to handle an expected peak of **400,000** users (1000 users per server × 400 servers), but unexpected positive press drives **500,000** users **in the first hour**.
- Based on the 100,000 who are turned away or receive bad service, by our assumption 10,000 are permanently lost, leaving an active user base of **390,000**.
- The **next hour** sees **250,000 new unique users**. The first 10,000 do fine, but the site is still over capacity by 240,000 users.
- This results in **24,000 additional defections**, leaving **376,000 permanent users**.
- If this pattern continues, after $\log_2(500000)$ or **19 hours**, the number of new users will approach zero and the site will be at capacity in steady state.

The service operator has collected **less than 400,000 users' worth of steady revenue** during those 19 hours, resulting in:

- Underutilization and
- bad reputation.

New Users	Users seeking service	Users turned away or receive bad service	Defectors	Active users
	0	0		0
500000	500000	100000	10000	390000
250000	640000	240000	24000	376000
125000	501000	101000	10100	389900
62500	452400	52400	5240	394760
31250	426010	26010	2601	397399
15625	413024	13024	1302	398698
7813	406510	6510	651	399349
3906	403255	3255	326	399674
1953	401628	1628	163	399837
977	400814	814	81	399919
488	400407	407	41	399959
244	400203	203	20	399980
122	400102	102	10	399990
61	400051	51	5	399995
31	400025	25	3	399997
15	400013	13	1	399999
8	400006	6	1	399999
4	400003	3	0	400000
2	400002	2	0	400000
1	400001	1	0	400000



(c) Underprovisioning 2

- Scale-up elasticity can be an **operational requirement**
- Scale-down elasticity allowed the **steady-state expenditure** to more closely **match** the steady-state **workload**.

Key benefit of Cloud Computing

The risk of mis-estimating workload is shifted from the service operator to the cloud vendor.

Cloud or Cluster? Cost-benefit Analysis

- Assumptions:
  - Cloud Computing vendor employs usage-based pricing
  - Customers pay proportionally to the amount of time and the amount of resources they use.
  - **Customer's revenue** is directly proportional to the **total number of user-hours**.
- When does it make sense to use Cloud vs local cluster?

"Above the Clouds: A Berkeley View of Cloud Computing", Armbrust et al. TR No. UCB/EECS-2009-28, 2009.

ARE THESE ASSUMPTIONS REALISTIC AND FEASIBLE FINANCIALLY?

The Economics of Cloud Computing

- Assumption is consistent with the ad-supported revenue model:
  - number of ads served roughly proportional to the total visit time spent by end users on a service.
- "Advertisers routinely pay more than a dollar per thousand impressions (**CPM**).
- If Google or Hotmail can collect a dollar per CPM, the resulting billion dollars per year will more than pay for their development and operating expenses.
- If t hey can deliver a search or a mail message for a few micro-dollars, the advertising pays them a few milli-dollars for the incidental "eyeballs".

Distributed Computing Economics, Jim Gray, Microsoft-TR-2003-24, 2003.

Cost-benefit Analysis

$$\boxed{\text{UserHours}_{\text{cloud}} \times (\text{revenue} - \text{Cost}_{\text{cloud}})} \geq \boxed{\text{UserHours}_{\text{datacenter}} \times \left(\text{revenue} - \frac{\text{Cost}_{\text{datacenter}}}{\text{Utilization}}\right)}$$

- Left-hand side: multiplies the net revenue per user-hour (revenue realized per user-hour minus cost of paying Cloud Computing per user-hour) by the number of user-hours, giving the expected profit from using Cloud Computing.
- Right-hand side: performs the same calculation for a fixed-capacity datacenter by *factoring in the average utilization*, including non-peak workloads.
- Whichever side is greater represents the opportunity for higher profit.

Cloud or Cluster? Cost-benefit Analysis

$$\text{UserHours}_{\text{cloud}} \times (\text{revenue} - \text{Cost}_{\text{cloud}}) \geq \text{UserHours}_{\text{datacenter}} \times \left(\text{revenue} - \frac{\text{Cost}_{\text{datacenter}}}{\text{Utilization}}\right)$$

- If the data center utilization equals 1, then the two sides of the inequality look the same. However:
  - As utilisation ->**1**, system response -> **infinity** (queuing theory result), so:
  - **Usable capacity** of a data center cluster is **0.6-0.8**; beyond this you cannot provide acceptable service => you need to over-provision your DC
- Key factor: cost per user hour of operating the service.

"Above the Clouds: A Berkeley View of Cloud Computing", Armbrust et al. TR No. UCB/EECS-2009-28, 2009.

Cloud or cluster?

Benefits of cloud computing over owning clusters:

$$\text{UserHours}_{\text{cloud}} \times (\text{revenue} - \text{Cost}_{\text{cloud}}) \geq \text{UserHours}_{\text{datacenter}} \times \left(\text{revenue} - \frac{\text{Cost}_{\text{datacenter}}}{\text{Utilization}}\right)$$

- Without elasticity, cost is high because resources sit idle.
- Underestimating spikes, turns users away and some leave for ever: fixed cost stay the same but amortized over fewer user-hours.
- For bursty workloads, utility computing makes more sense!
- Unexpected scale-down of own infrastructure (decommissioning) results in financial penalty.
- Hardware costs fall and savings can pass to cloud customers who can benefit from this without incurring a capital expense.

Amazon EC2 pricing models

aws academy

On-Demand Instances
- Pay by the hour
- No long-term commitments.
- Eligible for the AWS Free Tier.

Dedicated Hosts
- A physical server with EC2 instance capacity fully dedicated to your use.

Dedicated Instances
- Instances that run in a VPC on hardware that is dedicated to a single customer.

Reserved Instances
- Full, partial, or no upfront payment for instance you reserve.
- Discount on hourly charge for that instance.
- 1-year or 3-year term.

Scheduled Reserved Instances
- Purchase a capacity reservation that is always available on a recurring schedule you specify.
- 1-year term.

Spot Instances
- Instances run as long as they are available and your bid is above the Spot Instance price.
- They can be interrupted by AWS with a 2-minute notification.
- Interruption options include terminated, stopped or hibernated.
- Prices can be significantly less expensive compared to On-Demand Instances
- Good choice when you have flexibility in when your applications can run.

Per second billing available for On-Demand Instances, Reserved Instances, and Spot Instances that run Amazon Linux or Ubuntu.

Amazon EC2 pricing models: Benefits

aws academy

On-Demand Instances	Spot Instances	Reserved Instances	Dedicated Hosts
Low cost and flexibility	Large scale, dynamic workload	Predictability ensures compute capacity is available when needed	Save money on licensing costs Help meet compliance and regulatory requirements

Amazon EC2 pricing models: Use cases

aws academy

Spiky Workloads	Time-Insensitive Workloads	Steady-State Workloads	Highly Sensitive Workloads

On-Demand Instances	Spot Instances	Reserved Instances	Dedicated Hosts
• Short-term, spiky, or unpredictable workloads • Application development or testing	• Applications with flexible start and end times • Applications only feasible at very low compute prices • Users with urgent computing needs for large amounts of additional capacity	• Steady state or predictable usage workloads • Applications that require reserved capacity, including disaster recovery • Users able to make upfront payments to reduce total computing costs even further	• Bring your own license (BYOL) • Compliance and regulatory restrictions • Usage and licensing tracking • Control instance placement

Knowledge Check

- Explain what is horizontal and vertical elasticity, how they differ and which one is more difficult to achieve.
- Explain what overprovisioning means and discuss if it is a problem and why.

SHOULD I MOVE MY BUSINESS TO THE CLOUD?

University of Cyprus
Department of Computer Science

M. D. Dikaiakos

Cloud Computing: Key Features

The benefits of moving to the Cloud

University of Cyprus
Department of Computer Science

Should I move my business to the Cloud?

- Is it more economical to move existing datacenter-hosted service to the cloud, or keep it in a datacenter?
- Things to consider: Hardware cost evolution

Should I move to the Cloud?

	WAN bandwidth/mo.
Item in 2003	1 Mbps WAN link
Cost in 2003	$100/mo.
$1 buys in 2003	1 GB
Item in 2008	100 Mbps WAN link
Cost in 2008	$3600/mo.
$1 buys in 2008	2.7 GB
cost/performance improvement	2.7x

Should I move to the Cloud?

	WAN bandwidth/mo.	CPU hours (all cores)
Item in 2003	1 Mbps WAN link	2 GHz CPU, 2 GB DRAM
Cost in 2003	$100/mo.	$2000
$1 buys in 2003	1 GB	8 CPU hours
Item in 2008	100 Mbps WAN link	2 GHz, 2 sockets, 4 cores/socket, 4 GB DRAM
Cost in 2008	$3600/mo.	$1000
$1 buys in 2008	2.7 GB	128 CPU hours
cost/performance improvement	2.7x	16x

wide-area networking costs have improved the least in 5 years, by less than a factor of 3

Should I move to the Cloud?

	WAN bandwidth/mo.	CPU hours (all cores)	disk storage
Item in 2003	1 Mbps WAN link	2 GHz CPU, 2 GB DRAM	200 GB disk, 50 Mb/s transfer rate
Cost in 2003	$100/mo.	$2000	$200
$1 buys in 2003	1 GB	8 CPU hours	1 GB
Item in 2008	100 Mbps WAN link	2 GHz, 2 sockets, 4 cores/socket, 4 GB DRAM	1 TB disk, 115 MB/s sustained transfer
Cost in 2008	$3600/mo.	$1000	$100
$1 buys in 2008	2.7 GB	128 CPU hours	10 GB
cost/performance improvement	2.7x	16x	10x

Computing costs have improved the most in 5 years

However, the ability to use the extra computing power is based on the assumption that programs can utilize all the cores on both sockets in the computer.

This assumption is likely more true for Utility Computing, with many VMs serving thousands to millions of customers, than it is for programs inside the datacenter of a single company

Should I move to the Cloud?

	WAN bandwidth/mo.	CPU hours (all cores)	disk storage
Item in 2003	1 Mbps WAN link	2 GHz CPU, 2 GB DRAM	200 GB disk, 50 Mb/s transfer rate
Cost in 2003	$100/mo.	$2000	$200
$1 buys in 2003	1 GB	8 CPU hours	1 GB
Item in 2008	100 Mbps WAN link	2 GHz, 2 sockets, 4 cores/socket, 4 GB DRAM	1 TB disk, 115 MB/s sustained transfer
Cost in 2008	$3600/mo.	$1000	$100
$1 buys in 2008	2.7 GB	128 CPU hours	10 GB
cost/performance improvement	2.7x	16x	10x
Cost to rent $1 worth on AWS in 2008	**$0.27–$0.40** ($0.10–$0.15/GB × 3 GB)	**$2.56** (128× 2 VM's@$0.10 each)	**$1.20–$1.50** ($0.12–$0.15/GB-month × 10 GB)

At first glance, it appears that a given dollar will go further if used to purchase hardware in 2008 than to pay for use of that same hardware.

Other Factors to Consider

- Paying separate per resource or not?
  - Most applications do not make equal use of computation, storage, and network bandwidth;
  - Some are CPU-bound, others network-bound, and so on, and may saturate one resource while underutilizing others.
  - Pay-as-you-go Cloud Computing can charge the application separately for each type of resource, reducing the waste of underutilization.
- Power, cooling and amortized building costs should be considered:
  - Some estimates that the costs of CPU, storage and bandwidth roughly double when those costs are amortized over a building's lifetime.
- Operations costs: managing software upgrades, fault detection & fix
- Software complexity and cost of migration

Public Cloud or Private Data Center?

Table 1. Comparing public clouds and private data centers.

Advantage	Public Cloud	Conventional Data Center
Appearance of infinite computing resources on demand	Yes	No

M. Armbrust et al., ***"A view of cloud computing,"*** *Communications of the ACM, vol. 53, no. 4, p. 50, 2010*

Public Cloud or Private Data Center?

Table 1. Comparing public clouds and private data centers.

Advantage	Public Cloud	Conventional Data Center
Appearance of infinite computing resources on demand	Yes	No
Elimination of an up-front commitment by Cloud users	Yes	No

M. Armbrust et al., ***"A view of cloud computing,"*** *Communications of the ACM, vol. 53, no. 4, p. 50, 2010*

Public Cloud or Private Data Center?

Table 1. Comparing public clouds and private data centers.

Advantage	Public Cloud	Conventional Data Center
Appearance of infinite computing resources on demand	Yes	No
Elimination of an up-front commitment by Cloud users	Yes	No
Ability to pay for use of computing resources on a short-term basis as needed	Yes	No

M. Armbrust et al., ***"A view of cloud computing,"*** *Communications of the ACM, vol. 53, no. 4, p. 50, 2010*

Public Cloud or Private Data Center?

Table 1. Comparing public clouds and private data centers.

Advantage	Public Cloud	Conventional Data Center
Appearance of infinite computing resources on demand	Yes	No
Elimination of an up-front commitment by Cloud users	Yes	No
Ability to pay for use of computing resources on a short-term basis as needed	Yes	No
Economies of scale due to very large data centers	Yes	Usually not

M. Armbrust et al., ***"A view of cloud computing,"*** *Communications of the ACM, vol. 53, no. 4, p. 50, 2010*

Public Cloud or Private Data Center?

Table 1. Comparing public clouds and private data centers.

Advantage	Public Cloud	Conventional Data Center
Appearance of infinite computing resources on demand	Yes	No
Elimination of an up-front commitment by Cloud users	Yes	No
Ability to pay for use of computing resources on a short-term basis as needed	Yes	No
Economies of scale due to very large data centers	Yes	Usually not
Higher utilization by multiplexing of workloads from different organizations	Yes	Depends on company size

University of Cyprus
Department of Computer Science

M. Armbrust et al., ***"A view of cloud computing,"*** *Communications of the ACM, vol. 53, no. 4, p. 50, 2010*

Public Cloud or Private Data Center?

Table 1. Comparing public clouds and private data centers.

Advantage	Public Cloud	Conventional Data Center
Appearance of infinite computing resources on demand	Yes	No
Elimination of an up-front commitment by Cloud users	Yes	No
Ability to pay for use of computing resources on a short-term basis as needed	Yes	No
Economies of scale due to very large data centers	Yes	Usually not
Higher utilization by multiplexing of workloads from different organizations	Yes	Depends on company size
Simplify operation and increase utilization via resource virtualization	Yes	No

University of Cyprus
Department of Computer Science

M. Armbrust et al., ***"A view of cloud computing,"*** *Communications of the ACM, vol. 53, no. 4, p. 50, 2010*

Public Cloud or Private Data Center?

Table 1. Comparing public clouds and private data centers.

Advantage	Public Cloud	Conventional Data Center
Appearance of infinite computing resources on demand	Yes	No
Elimination of an up-front commitment by Cloud users	Yes	No
Ability to pay for use of computing resources on a short-term basis as needed	Yes	No
Economies of scale due to very large data centers	Yes	Usually not
Higher utilization by multiplexing of workloads from different organizations	Yes	Depends on company size
Simplify operation and increase utilization via resource virtualization	Yes	No

University of Cyprus
Department of Computer Science

M. Armbrust et al., ***"A view of cloud computing,"*** *Communications of the ACM, vol. 53, no. 4, p. 50, 2010*

The paradox of Cloud

- "while cloud clearly delivers on its promise early on in a company's journey, the pressure it puts on margins can start to outweigh the benefits, as the company scales and growth slows."
- Repatriation from the cloud results in 1/3 to 1/2 the cost of running equivalent workloads in the cloud.
- Public cloud list prices can be 10 to 12x the cost of running one's own data centers.

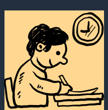
You're crazy if you don't start in the cloud; you're crazy if you stay on it.

- Companies need to optimize infrastructure spending early, often, and, sometimes, also outside the cloud.

University of Cyprus
Department of Computer Science

Source: The Cost of Cloud, a Trillion Dollar Paradox, S. Wang & M. Casado, Andreessen Horowitz, 27/5/2021

Questions for Homework



CAN YOU UPDATE TABLE 5 (GRAY'S UPDATED ESTIMATES) WITH DATA FOR 2022 (US?)

AN ORGANISATION IS PLANNING TO SPEND €1M TO COMMISSION A LARGE CLUSTER. CAN YOU PROVIDE SOME ALTERNATIVE SCENARIOS OF USAGE, TO BE SUPPORTED BY THIS AMOUNT OF FUNDING?

University of Cyprus
Department of Computer Science

M. D. Dikaiakos

Homework

- Suppose a biology lab creates 1 TB of new data for every wet lab experiment.
- A computer the speed of one EC2 instance takes 1 hour per GB to process the new data.
- The lab has the equivalent 5 instances locally.
- Explore the tradeoffs between computing the experiments in house, on Amazon, on Google Cloud, on Azure or on a local (Cypriot) Cloud provider.

University of Cyprus
Department of Computer Science

M. D. Dikaiakos

Cloud Computing: Key Features

Cloud Resource Virtualization

Questions about Resources

Which resources can be offered as utilities?

- Computing
  - CPU cores, server nodes, clusters, accelerators
- Storage
  - Cache, RAM, disk space, file system space, database records
- Communication
  - Network topology, number of messages, messages per second, guaranteed latency, bandwidth
- Application-oriented
  - Function calls, queries, transactions, file reads/writes, other

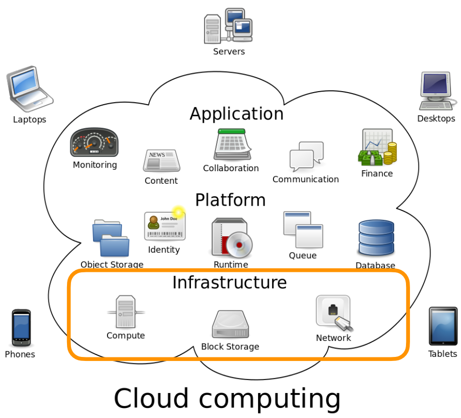
Questions about Resources

- **How** are resources exposed to the Cloud user / application developer?
- How can the application developer take advantage of or implement elasticity mechanisms?
  - Level of management and tuning required?
  - Abstractions offered?

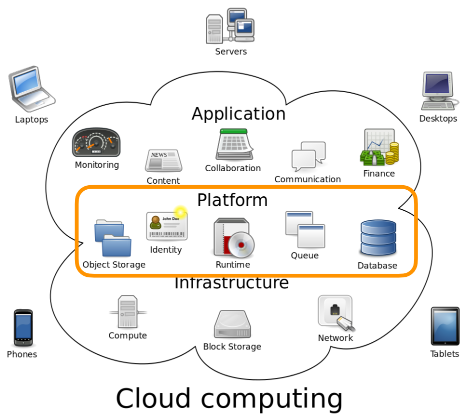
Resource Virtualization

- The statistical multiplexing necessary to achieve elasticity and the illusion of infinite capacity requires resources to be **virtualized** so that:
- the implementation of how they are multiplexed and shared can be hidden (abstracted) from the programmer.

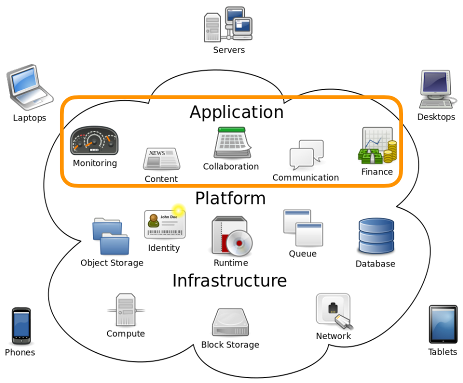
Abstraction Layers



Abstraction Layers



Abstraction Layers



Cloud computing

Cloud Computing: Key Features

Profile of a Cloud Provider

WHO CAN BECOME A CLOUD PROVIDER?

Requirements

- Very Large Investments (necessary, not sufficient) in
  - very large data centers,
  - large-scale software infrastructure,
  - operational expertise to run them.
- Building, provisioning, launching: **$100M** undertaking.
  - Amazon, eBay, Google, Microsoft (early 2000's)

Becoming a Cloud provider

Economies of scale

- Make a lot of money: a sufficiently large company could leverage **economies of scale** to offer a service well below the costs of a medium-sized company and still make a tidy profit
- Very large data centers (tens of thousands of computers) can purchase hardware, network bandwidth, and power for **1/5 to 1/7** the prices offered to a medium-sized (hundreds or thousands of computers) data center.
- The fixed costs of software development and deployment can be amortized over many more machines.
- Others estimate the price advantage as a factor of **3 to 5**.

Economies of Scale

Table 2: Economies of scale in 2006 for medium-sized datacenter (≈1000 servers) vs. very large datacenter (≈50,000 servers). [24]

Technology	Cost in Medium-sized DC	Cost in Very Large DC	Ratio
Network	$95 per Mbit/sec/month	$13 per Mbit/sec/month	7.1
Storage	$2.20 per GByte / month	$0.40 per GByte / month	5.7
Administration	≈140 Servers / Administrator	>1000 Servers / Administrator	7.1

Becoming a Cloud: factors

Leverage existing investment

- Adding Cloud Computing services on top of existing infrastructure provides a new revenue stream at (ideally) low incremental cost, helping to amortize the large investments of data centers.

Defend a franchise.

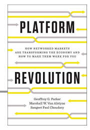
- As conventional server and enterprise applications embraced Cloud Computing, vendors with an established franchise in those applications would be motivated to provide a cloud option of their own (e.g. Microsoft Azure).

Attack an incumbent.

- A company with the requisite datacenter and software resources might want to establish a beachhead in this space before a single "800 pound gorilla" emerges.
- Google AppEngine provides an alternative path to cloud deployment whose appeal lies in its automation of many of the scalability and load balancing features that developers might otherwise have to build for themselves.

Becoming a Cloud Provider

- **Leverage customer relationships.**
  - IT service organisations such as IBM have extensive customer relationships through their service offerings.
  - Providing a branded Cloud Computing offering gives those customers an anxiety-free migration path that preserves both parties' investments in the customer relationship.
- **Become a platform**.
  - Facebook's plug-in apps: a great fit for cloud computing.
  - Facebook's motivation: turn their social-networking application a new development platform.

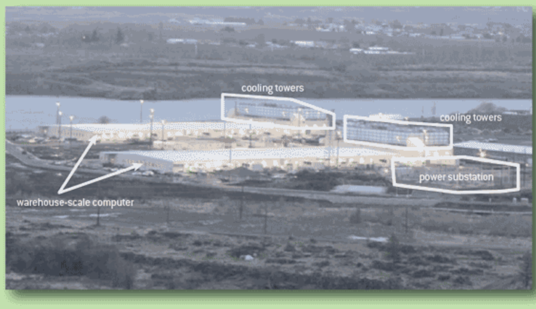


Cloud Computing: Key Features

Cloud Data Centers

Source: Luiz André Barroso, Urs Hölzle, and Parthasarathy Ranganatha, "The Datacenter as a Computer: Designing Warehouse-Scale Machines," Third Edition Morgan & Claypool (2019).

FIGURE 1 An Example Data Center and Warehouse-Scale Computer



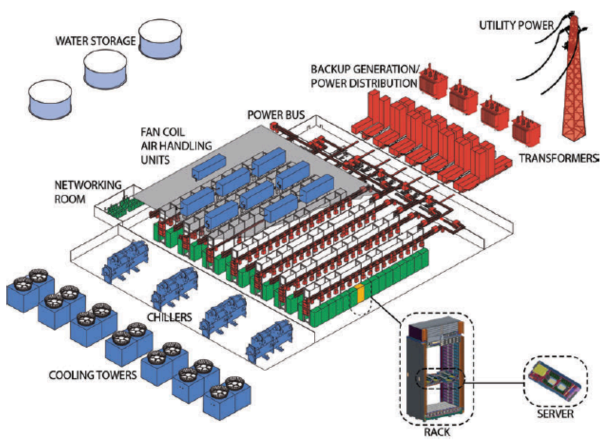
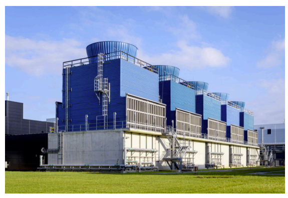
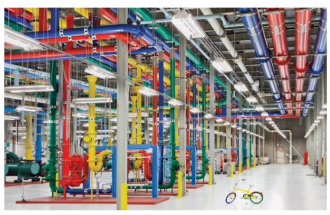


Figure 4.4: The main components of a typical data center.



: Power distribution, Council Bluffs, Iowa, U.S.



Data center cooling, Douglas County, Georgia, U.S.

Cloud Data Centers - WSC

- Belong to a single organization.
- Use a relatively homogeneous hardware and system software platform.
- Share a common systems management layer: Application, middleware, and system software is built in-house, allows significant deployment flexibility.
- Run a smaller number of very large applications (or Internet services).

Requirements

- High Availability (at least 99.99% uptime - about an hour of downtime per year).
  - Fault-free operation possible but extremely expensive.
- Cost-Efficiency: a primary metric of interest in the design of WSCs (why?).

Cloud Data Centers

Data Center Location

Location Criteria

- **Cost:** electricity, cooling, labor, land property, and taxes (geographically variable).
  - **Electricity** and **cooling:** ~**1/3 of the costs**.
  - Physics tells us: it's easier (cheaper) to ship photons than electrons.
- **Proximity** to large urban centers:
  - determines latency
- **Policy issues**

Source: Armbrust, A. Fox, and R. Griffith, M. (2009). Above the clouds: A Berkeley view of cloud computing. In University of California, Berkeley, TR 2009-28.

Table 3: Price of kilowatt-hours of electricity by region [7].

Price per KWH	Where	Possible Reasons Why
3.6¢	Idaho	Hydroelectric power; not sent long distance
10.0¢	California	Electricity transmitted long distance over the grid; limited transmission lines in Bay Area; no coal fired electricity allowed in California.
18.0¢	Hawaii	Must ship fuel to generate electricity

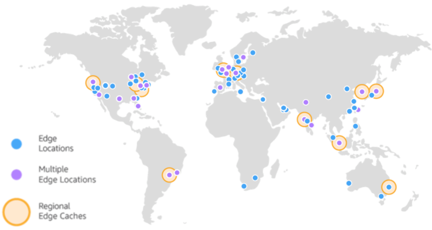
Source: Armbrust, A. Fox, and R. Griffith, M. (2009). Above the clouds: A Berkeley view of cloud computing. In University of California, Berkeley, TR 2009-28.

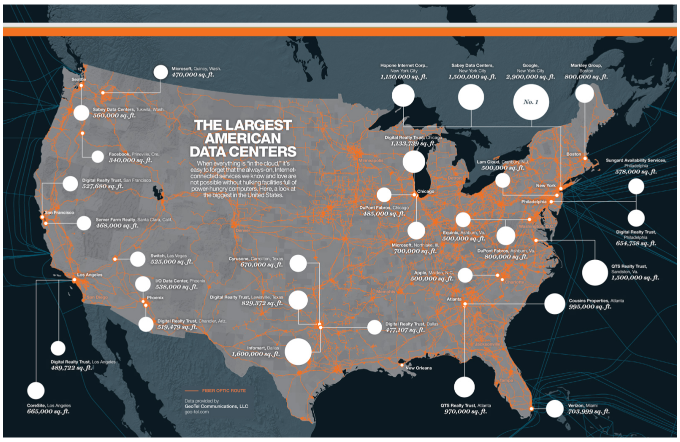
Figure 1.7: Google Cloud Platform regions and number of zones, circa July 2018. The latest source is available at https://cloud.google.com/about/locations/.

Points of Presence

aws academy

- AWS provides a global network of 187 Points of Presence locations
- Consists of 176 **edge locations** and 11 **Regional edge caches**
- Used with Amazon CloudFront
  - A global Content Delivery Network (CDN), that delivers content to end users with reduced latency
- Regional edge caches used for content with infrequent access.





Circa 2014

http://www.iiclouds.org/20141114/maps-of-data-center-localization/

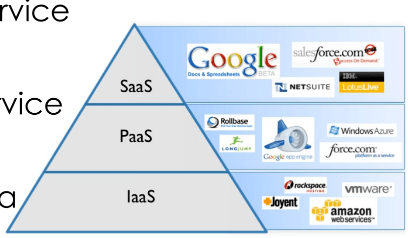
Classification of Cloud Offerings

Cloud Service Models

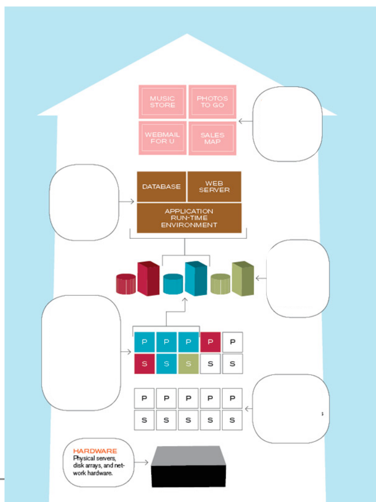
Based on Service Models

What kind of Cloud service is provided as a utility offering?

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



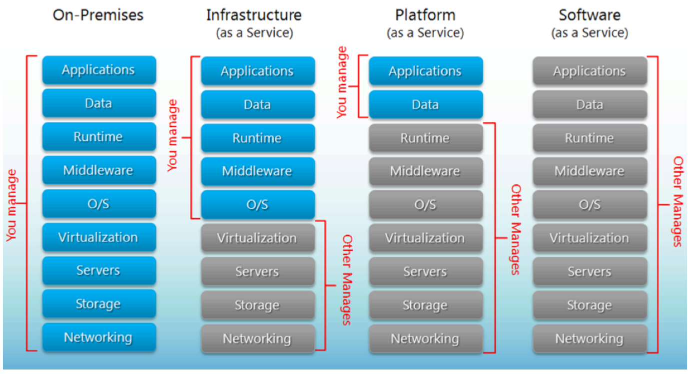
The NIST Definition of Cloud Computing, NIST, 2011

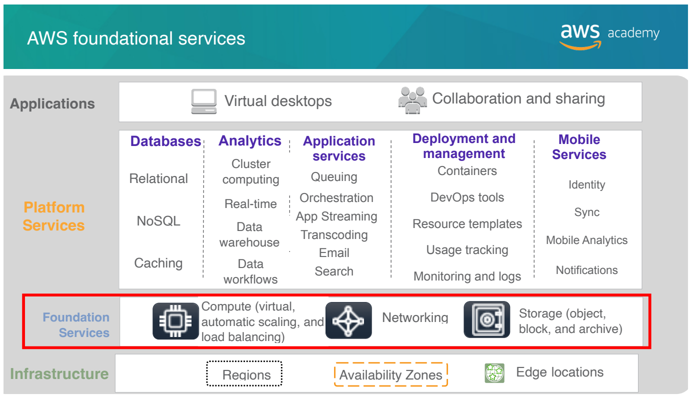


aws academy

IaaS (infrastructure as a service)	PaaS (platform as a service)	SaaS (software as a service)

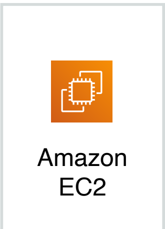
More control over IT resources → Less control over IT resources





Amazon EC2 overview

aws academy



- **Amazon Elastic Compute Cloud (Amazon EC2)**
  - Provides virtual machines—referred to as EC2 instances—in the cloud.
  - Gives you *full control* over the guest operating system (Windows or Linux) on each instance.
- You can launch instances of any size into an Availability Zone anywhere in the world.
  - Launch instances from **Amazon Machine Images (AMIs)**.
  - Launch instances with a few clicks or a line of code, and they are ready in minutes.
- You can control traffic to and from instances.

Flexibility — **Programming Convenience**

Amazon EC2

- Looks like physical hardware
- Users can control nearly the entire software stack, from the kernel upwards.
- Exposed API is "thin"
- No prior limit on applications that can be hosted
- Allows developers to code whatever they want

- Difficult to offer automatic scalability and failover, because the semantics associated with replication and other state management issues are highly application-dependent.

MS Azure

- The system supports general-purpose computing, rather than a single category of application.
- Users get a choice of language, but cannot control the underlying operating system or runtime.
- .NET libraries provide a degree of automatic network configuration and failover/scalability, but
- require the developer to declaratively specify some application properties in order to do so.

Google AppEngine, SalesForce

- Targeted exclusively at traditional web applications, enforcing an application structure of clean separation between a *stateless* computation tier and a *stateful* storage tier.
- Applications expected to be request-reply, and severely rationed in how much CPU time they can use in servicing a particular request.
- Automatic scaling and high-availability mechanisms, and the proprietary MegaStore data storage available to AppEngine applications, all rely on these constraints.
- Force.com is designed to support business applications that run against the salesforce.com database, and nothing else.
- Not suitable for general-purpose computing.

WHICH SERVICE MODEL IS BETTER?

Table 4: Examples of Cloud Computing vendors and how each provides virtualized resources (computation, storage, networking) and ensures scalability and high availability of the resources.

	Amazon Web Services	Microsoft Azure	Google AppEngine
Computation model (VM)	• x86 Instruction Set Architecture (ISA) via Xen VM • Computation elasticity allows scalability, but developer must build the machinery, or third party VAR such as RightScale must provide it	• Microsoft Common Language Runtime (CLR) VM; common intermediate form executed in managed environment • Machines are provisioned based on declarative descriptions (e.g. which "roles" can be replicated); automatic load balancing	• Predefined application structure and framework; programmer-provided "handlers" written in Python, all persistent state stored in MegaStore (outside Python code) • Automatic scaling up and down of computation and storage; network and server failover; all consistent with 3-tier Web app structure

Table 4: Examples of Cloud Computing vendors and how each provides virtualized resources (computation, storage, networking) and ensures scalability and high availability of the resources.

	Amazon Web Services	Microsoft Azure	Google AppEngine
Computation model (VM)	• x86 Instruction Set Architecture (ISA) via Xen VM • Computation elasticity allows scalability, but developer must build the machinery, or third party VAR such as RightScale must provide it	• Microsoft Common Language Runtime (CLR) VM; common intermediate form executed in managed environment • Machines are provisioned based on declarative descriptions (e.g. which "roles" can be replicated); automatic load balancing	• Predefined application structure and framework; programmer-provided "handlers" written in Python, all persistent state stored in MegaStore (outside Python code) • Automatic scaling up and down of computation and storage; network and server failover; all consistent with 3-tier Web app structure
Storage model	• Range of models from block store (EBS) to augmented key/blob store (SimpleDB) • Automatic scaling varies from no scaling or sharing (EBS) to fully automatic (SimpleDB, S3), depending on which model used • Consistency guarantees vary widely depending on which model used • APIs vary from standardized (EBS) to proprietary	• SQL Data Services (restricted view of SQL Server) • Azure storage service	•MegaStore/BigTable

Table 4: Examples of Cloud Computing vendors and how each provides virtualized resources (computation, storage, networking) and ensures scalability and high availability of the resources.

	Amazon Web Services	Microsoft Azure	Google AppEngine
Computation model (VM)	• x86 Instruction Set Architecture (ISA) via Xen VM • Computation elasticity allows scalability, but developer must build the machinery, or third party VAR such as RightScale must provide it	• Microsoft Common Language Runtime (CLR) VM; common intermediate form executed in managed environment • Machines are provisioned based on declarative descriptions (e.g. which "roles" can be replicated); automatic load balancing	• Predefined application structure and framework; programmer-provided "handlers" written in Python, all persistent state stored in MegaStore (outside Python code) • Automatic scaling up and down of computation and storage; network and server failover; all consistent with 3-tier Web app structure
Storage model	• Range of models from block store (EBS) to augmented key/blob store (SimpleDB) • Automatic scaling varies from no scaling or sharing (EBS) to fully automatic (SimpleDB, S3), depending on which model used • Consistency guarantees vary widely depending on which model used • APIs vary from standardized (EBS) to proprietary	• SQL Data Services (restricted view of SQL Server) • Azure storage service	•MegaStore/BigTable
Networking model	• Declarative specification of IP-level topology; internal placement details concealed • Security Groups enable restricting which nodes may communicate • Availability zones provide abstraction of independent network failure • Elastic IP addresses provide persistently routable network name	• Automatic based on programmer's declarative descriptions of app components (roles)	• Fixed topology to accommodate 3-tier Web app structure • Scaling up and down is automatic and programmer-invisible

Classification of Cloud Offerings

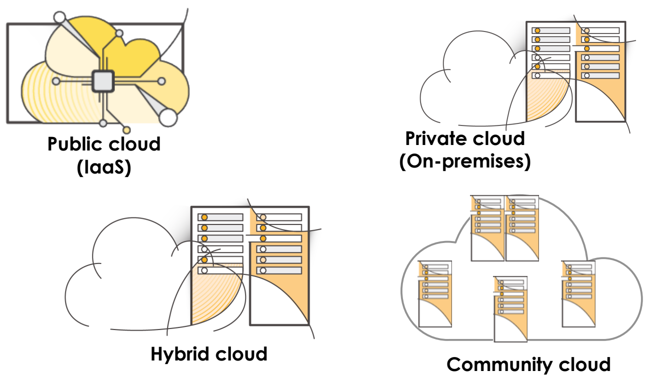
Cloud Deployment Models

Based on ownership

Who owns/controls the resources where cloud utility offerings are deployed?

The NIST Definition of Cloud Computing, NIST, 2011

Deployment Models



The NIST Definition of Cloud Computing, NIST, 2011

WHICH DEPLOYMENT MODEL IS BETTER?

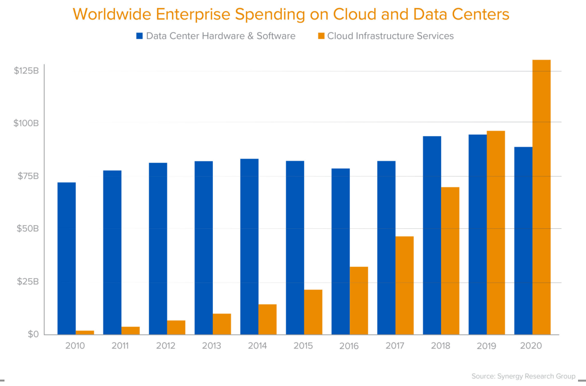
Obstacles and Opportunities

Table 2. Top 10 obstacles to and opportunities for growth of cloud computing.

	Obstacle	Opportunity
1	Availability/Business Continuity	Use Multiple Cloud Providers
2	Data Lock-In	Standardize APIs; Compatible SW to enable Surge or Hybird Cloud Computing
3	Data Confidentiality and Auditability	Deploy Encryption, VLANs, Firewalls
4	Data Transfer Bottlenecks	FedExing Disks; Higher BW Switches
5	Performance Unpredictability	Improved VM Support; Flash Memory; Gang Schedule VMs
6	Scalable Storage	Invent Scalable Store
7	Bugs in Large Distributed Systems	Invent Debugger that relies on Distributed VMs
8	Scaling Quickly	Invent Auto-Scaler that relies on ML; Snapshots for Conservation
9	Reputation Fate Sharing	Offer reputation-guarding services like those for email
10	Software Licensing	Pay-for-use licenses

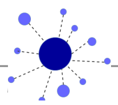
Adoption Growth Policy & Business

Public Clouds vs Private Data Centers



Public Clouds: Challenges & Concerns

- Emerging applications:
  - Increasing complexity & Dynamic behavior
  - A variety of deployment platforms with different: configuration mechanisms, offered services, availability and pricing, elasticity capabilities, "devops" requirements
- A world of Silos:
  - Lack of interoperability, diversity in technology offerings, APIs, policies, vendor-specific tools, etc
  - Restricted portability, migration entails significant cost
  - User lock-in by design?



Cloud Computing: Introduction and Key Concepts

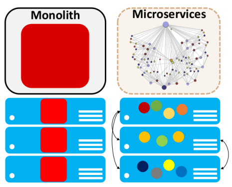
Cloud Computing Application Models

New Applications on the Cloud

- Economic necessity mandates putting the data near the application [J. Gray, 2003]
- Mobile interactive applications
- Parallel batch processing on very large datasets - MapReduce, Hadoop
- Rise of business analytics - Spark, Storm, Flink
- Seamless extension of computing-intensive desktop applications

Distributed Computing Economics, Jim Gray, Microsoft-TR-2003-24, 2003.

Monoliths VS Microservices



Microservices:

- provide **composable software design** that simplifying and accelerating development
- Enable **programming language** and **framework heterogeneity**
- Simplify **correctness** and **performance debugging**, as bugs can be isolated in specific tiers

*While the entire Monolith is scaled out on multiple servers, microservices allow **individual components** of the end-to-end application to **be elastically scaled.***

New Application Opportunities

- Economic necessity mandates putting the data near the application [J. Gray, 2003]
- Why?
  - The cost of wide-area networking has fallen more slowly (and remains relatively higher) than all other IT hardware costs
  - With $1 you can accomplish much more computation than communication
- How do you combine data from multiple sites?
  - Push as much of the processing to the data sources as possible in order to filter the data early.
- Although hardware costs have changed since Gray's analysis, his idea of this "breakeven point" has not (keep an eye on Moore's Law)

Distributed Computing Economics, Jim Gray, Microsoft-TR-2003-24, 2003.

Mobile interactive applications

"*the future belongs to* ***services that respond in realtime*** *to information provided either by their users or by nonhuman sensors*"

[Tim O'Reilly, ~2008]

- Such services are attracted to the cloud because they must be highly available, and generally rely on large datasets that are most *conveniently hosted in large data centers*.
  - Especially for services that combine *two or more data sources* or other services, e.g., *mashups*.
- Disconnected operation not a significant obstacle to the appeal of mobile applications (addressed successfully in specific application domains)

Parallel batch processing

- Cloud Computing presents a unique opportunity for batch-processing and analytics jobs that analyze terabytes of data and can take hours to finish.
- If there is enough data parallelism you can take advantage of the cloud's "**cost associativity**".
- What is needed for data parallelism to materialize?
  - Programming abstractions such as **MapReduce** and **Hadoop**: hide the operational complexity of choreographing parallel execution across hundreds of Cloud Computing servers.
  - Cost/benefit analysis: moving large datasets into the cloud vs. potential speedup in the data analysis.

The rise of analytics

- A special case of compute-intensive batch processing is **business analytics**.
  - Originally, database industry dominated by transaction processing.
  - A growing share of computing resources is now spent on understanding customers, supply chains, buying habits, ranking, etc.
  - Decision support growing rapidly, shifting the resource balance in database processing from transactions to business analytics.

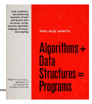
Extension of compute-intensive desktop applications

- The latest versions of the mathematics software packages Matlab and Mathematica are capable of using Cloud Computing to perform expensive evaluations.
- Other desktop applications might similarly benefit from seamless extension into the cloud.
- "Keep the data in the cloud and rely on having sufficient bandwidth to enable suitable visualization and a responsive GUI back to the human user."

"Earthbound" applications

- Some good candidate applications for the cloud may be thwarted by:
  - data movement costs
  - the fundamental latency limits of getting into and out of the cloud, or both.
- E.g.:
  - stock trading that requires microsecond precision is not appropriate for the Cloud
  - Sensor & Actuator applications on the Edge
  - Until the cost (and possibly latency) of wide-area data transfer decrease, such applications may be less obvious candidates for the cloud.

A Paradigm Shift



- Traditional programming model: **Algorithms + Data Structures = Programs**

- Cloud Computing application development signifies a departure from the traditional programming model where a program runs on a single machine.
- In warehouse-scale computing:
  - **Program**: an Internet service, which may consist of tens or more individual programs that interact to implement complex end-user services. These programs might be implemented and maintained by different teams of engineers, perhaps across organizational, geographic, and company boundaries.
  - **Computing platform** consists of thousands of individual computing nodes with their corresponding networking and storage subsystems, power distribution and air-conditioning equipment, and extensive cooling systems. The enclosure for these systems is a building structure.

Categorizing compute services

aws academy

Services	Key Concepts	Characteristics	Ease of Use
• Amazon EC2	• Infrastructure as a service (IaaS) • Instance-based • **Virtual machines**	• Provision virtual machines that you can manage as you choose	A familiar concept to many IT professionals.
• AWS Lambda	• **Serverless** computing • Function-based • Low-cost	• Write and deploy code that runs on a schedule or that can be triggered by events • Use when possible (architect for the cloud)	A relatively new concept for many IT staff members, but easy to use after you learn how.
• Amazon ECS • Amazon EKS • AWS Fargate • Amazon ECR	• **Container-based** computing • Instance-based	• Spin up and run jobs more quickly	AWS Fargate reduces administrative overhead, but you can use options that give you more control.
• AWS Elastic Beanstalk	• Platform as a service (PaaS) • For **web applications**	• Focus on your code (building your application) • Can easily tie into other services—databases, Domain Name System (DNS), etc.	Fast and easy to get started.

Reading Assignment

Read the following blog posts and summarize the key points each makes:

- Sarah Wang and Martin Casado, **"The Cost of Cloud, a Trillion Dollar Paradox"** https://a16z.com (5/2021)
- Martin Casado, **"The Cloud Killed Infrastructure, Long Live Infrastructure!"** https://a16z.com (6/2022)