



UMD DATA605 - Big Data Systems

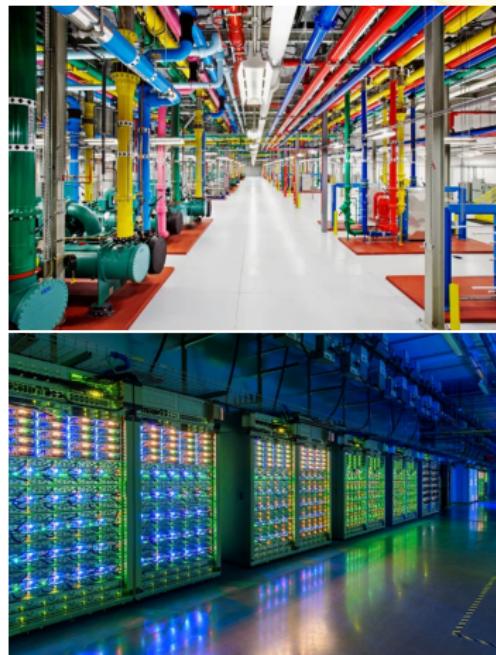
11.2: Technologies Enabling Cloud Computing

- **Instructor:** Dr. GP Saggese, gsaggese@umd.edu

- ***Data centers***
- Virtualization
- Programming frameworks
- Challenges and opportunities

Data Centers: Capex

- Data centers enable cloud computing
- Large companies (e.g., AWS, Apple, Google, Facebook) build data centers globally
- Data center costs around 1B USD to build
 - Capex: Computing, memory, storage, networking
 - Prices dropping
 - Size is increasing



Data Centers: Opex

- **Powering equipment cost**
 - Focus on energy-efficient computing
- **High cooling cost**
 - Vent placement is key to manage thermal hotspots
 - PUE (Power Usage Effectiveness)
 - Some power converted into computation
 - Rest is overhead
 - Hard to optimize in small data centers
 - Ideal PUE is 1
 - Current is 1.07-1.22
 - May lead to large data centers soon
- **Research on energy-saving**
- **Data centers built**
 - Close to cheap energy sources
 - In cold climate

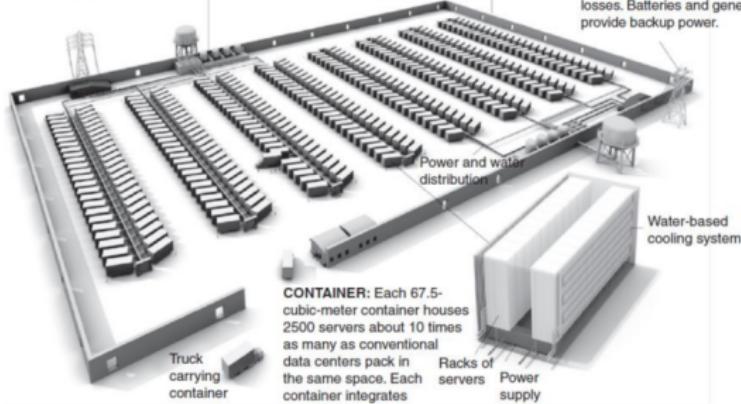


(Modular) Data Centers

COOLING: High-efficiency water-based cooling systems—less energy-intensive than traditional chillers—circulate cold water through the containers to remove heat, eliminating the need for air-conditioned rooms.

STRUCTURE: A 24 000-square-meter facility houses 400 containers. Delivered by trucks, the containers attach to a spine infrastructure that feeds network connectivity, power, and water. The data center has no conventional raised floors.

POWER: Two power substations feed a total of 300 megawatts to the data center, with 200 MW used for computing equipment and 100 MW for cooling and electrical losses. Batteries and generators provide backup power.



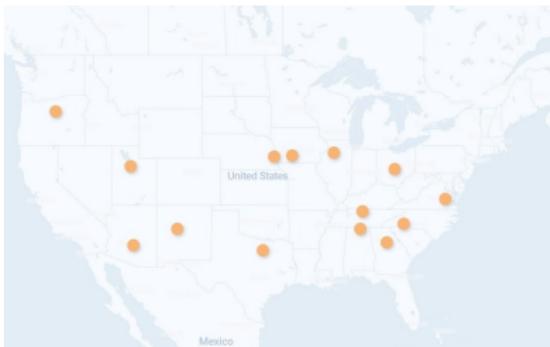
118 Dept of Computer Science UMD



Meta

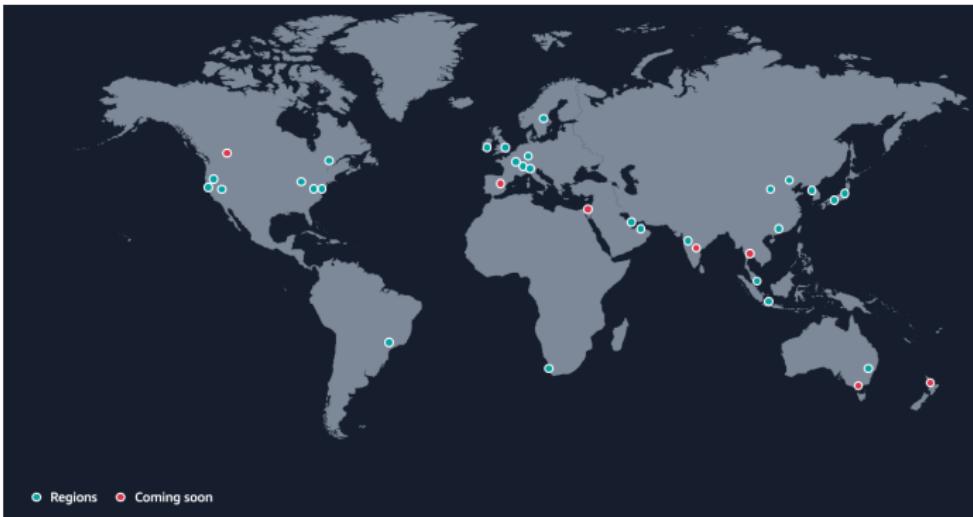
- Global scale of data centers
 - 29 dedicated Meta data centers
 - Hyperscale and AI-optimized facilities
- Investment
 - \$600 billion in U.S. infrastructure by 2028
 - El Paso data center: \$1.5 billion
 - Louisiana campus: \$10 billion for 9 buildings
 - Power infrastructure upgrades: over \$3 billion

Data Center	Online	Buildings	SqFt (m)	Investment (\$bn)
Dekalb, Illinois	2022	2	0.9	\$0.8
Altoona, Iowa	2014	10	4.1	\$2.0
Papillion (Sarpy), Nebraska	2019	8	3.6	\$1.5
New Albany, Ohio	2020	5	2.5	\$1.0
Huntsville, Alabama	2021	4	2.5	\$1.0
Newton, Georgia	2023	5	2.5	\$1.0
Forest City, North Carolina	2012	4	1.3	\$0.8
Gallatin, Tennessee	2023	2	1.0	\$0.8
Henrico, Virginia	2020	7	2.5	\$1.0
Mesa, Arizona	Q4 2023	2	1.0	\$0.8
Los Lunas, New Mexico	2019	6	2.8	\$1.0
Fort Worth, Texas	2017	5	2.6	\$1.5
Prineville, Oregon	2011	11	4.6	\$2.0
Eagle Mountain, Utah	2021	5	2.4	\$1.0
Odense, Denmark	2019	2	0.9	\$1.6
Clonee, Ireland	2018	3	1.6	\$0.4
Luleå, Sweden	2013	3	1.0	\$1.0
Tanjong Kling, Singapore	2022	1	1.8	\$1.0
Total		85	39.6	\$20.1



Amazon Web Services

- AWS
 - 2022: 28 geographical regions
 - 2025: 38 regions



Amazon Web Services (EC2)

- Widely used solution for cloud computing
 - Many alternatives to suit your needs
 - Prices are low due to competition
 - Current on-demand pricing

Small Instance – default*

1.7 GB memory
1 EC2 Compute Unit (1 virtual core with 1 EC2 Compute Unit)
160 GB instance storage
32-bit platform
I/O Performance: Moderate
API name: m1.small

Large Instance

7.5 GB memory
4 EC2 Compute Units (2 virtual cores with 2 EC2 Compute Units each)
850 GB instance storage
64-bit platform
I/O Performance: High
API name: m1.large

Extra Large Instance

15 GB memory
8 EC2 Compute Units (4 virtual cores with 2 EC2 Compute Units each)
1,690 GB instance storage
64-bit platform
I/O Performance: High
API name: m1.xlarge

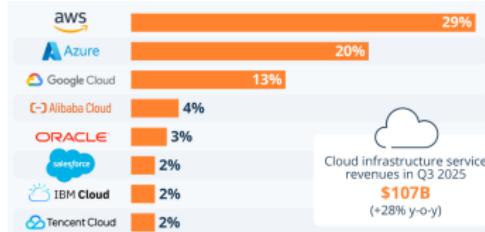
Amazon S3

- Amazon storage services (S3 = Simple Storage Solution)
 - Pay for storage you use
- Different tiers for reliability, cost, performance

	Default	RRS	IA	Glacier
Durability	99.999999999%	99.99%	99.999999999%	99.999999999%
Availability	99.99%	99.99%	99.9%	99.99%
Extra Fees	None	None	Retrieval	Retrieval
Real-Time Access?	Yes	Yes	Yes	No (mins/hours)
Frequently Accessed?	Yes	Yes	No	No

Google App Engine

- **Google Compute Engine** (IaaS)
 - Competes with AWS EC2
- **Google Infrastructure** (PaaS)
 - Run Docker containers on Google resources
 - Managed services (e.g., databases)
- **Google Docs** (SaaS)
 - Word processor, spreadsheet, presentations in the cloud
- Google Cloud's **compute market share**
 - Built software data centers before Amazon
 - Invented cloud technologies (e.g., Google File System, MapReduce, BigTable)
 - Market share 3x smaller than AWS
 - Issues:
 - Developer/customer unfriendliness
 - Lack of commitment (Killed by Google)
 - Poor customer service



- Data centers
- ***Virtualization***
- Programming frameworks
- Challenges and opportunities

Virtualization

- Virtual machines have been around for a long time
 - E.g., running Windows inside a Mac
 - Used to be slow (e.g., QEMU)
- Only recently (2000s) became efficient enough for cloud computing
- Basic idea of cloud computing
 - Run virtual machines on servers and sell time on them
 - E.g., amazon EC2, Microsoft Azure, Google Cloud
- Many advantages
 - Security: virtual machines have almost impenetrable boundary
 - Multi-tenancy: multiple VMs on the same server
 - Efficiency: replace many underpowered machines with fewer high-powered machines

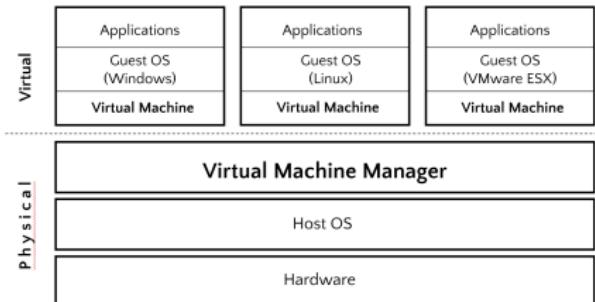
Virtualization

- **Consumer / desktop virtualization**

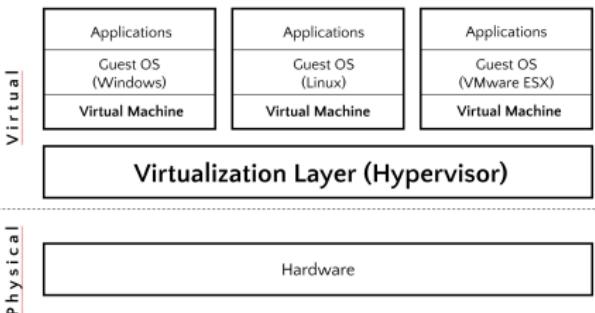
- VMWare, Xen, VirtualBox
- Run on host OS
- Hypervisor/VM supports guest OS

- **Server virtualization**

- Run hypervisor on hardware
- Ideal for server farms, cloud computing
- Amazon used Xen on RedHat
- Now uses AWS Nitro
- Supports Windows and Linux VMs



Consumer / desktop virtualization



Server virtualization

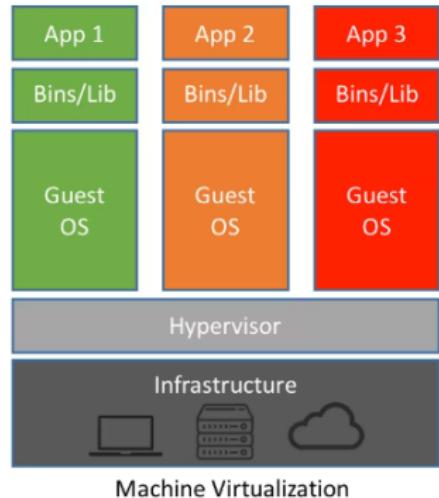
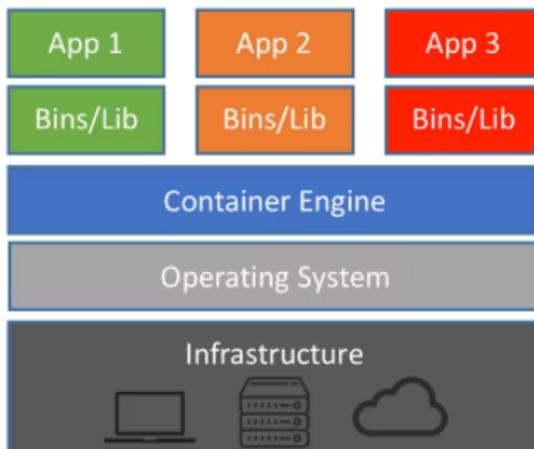


Virtualization

- Performance and tricky things to keep in mind
 - Hard to reason about performance
 - Identical VMs may deliver different performance
 - Multi-tenancy, different hardware
- “Bare-metal” compute to improve performance

Docker

- Enable true independence between application devs and IT ops
- Package all dependencies
- Advantages
 - Containers are fast and portable
 - Reduce virtualization overhead
 - Containers don't require full OS
 - All containers run on a single host
 - Reduce OS licensing cost
 - Reduce OS maintenance overhead



- Data centers
- Virtualization
- *Programming frameworks*
- Challenges and opportunities

Programming Frameworks

- Programming frameworks emerged from efforts to "scale out" workloads
 - Distribute work over thousands of machines
- Parallelism has been around for a long time
 - In a single machine and as a cluster of computers
- Parallelism is hard for programmers
 - Many things to track:
 - Parallelize application
 - Distribute data
 - Handle failures
 - Debugging
 - Race conditions/Heisenbugs
- The difference is the user interface
 - Google developed MapReduce and BigTable, starting a new era
 - Hadoop, Spark
 - AWS services

MapReduce Framework

- Provide a restricted, powerful abstraction for programming distributed workloads
- Programmers write functions: *map* and *reduce*
 - **map** programs
 - Input: list of “records” (e.g., images, genomes)
 - Output: set of (**key**, **value**) pairs for each record
 - **reduce** programs
 - Input: list of (**key**, [values]) from mapper
 - Output: flexible
 - Perform arbitrary computations on input data within structure
- Framework handles task scheduling, fault tolerance
- **Batch processing of data**
 - MapReduce and analytics frameworks excel
- **Streaming data**
 - Large-scale applications need real-time access with low latency
 - Balance “consistency” and “performance”

Other Programming Frameworks

- Many programming frameworks for different applications
 - Address MapReduce limitations
- **High-performance Computing (HPC) Systems**
 - Cluster of supercomputers
 - E.g., GridRPC, MPI
 - More expressive, efficient
- **Spark**
 - Based on Resilient Distributed Data (RDD)
 - In-memory, efficient
 - Uses Scala, Python, Java
- **Apache Hive**
 - SQL-like interface on Hadoop/HDFS
- **Apache HBase**
 - NoSQL column-oriented DB
 - Random read/write large tables on Hadoop/HDFS
 - Modeled after Google BigTable
- **Apache Storm, Spark Streaming**
 - Handle real-time streaming data
- **Giraph, GraphLab, GraphX**

SCIENCE • Graph processing systems
ACADEMY



- Data centers
- Virtualization
- Programming frameworks
- ***Challenges and opportunities***

Advantages of Cloud Computing

- **Lower edge computer costs**
 - Applications run in the cloud, reducing desktop processing power/memory/disk needs
- **Improved performance of desktop**
 - Faster system boot due to less memory usage by large programs
- **Device independence**
 - Not tethered to a single computer
 - Applications/documents follow you through the cloud
 - Access applications/documents on any device
- **Reduced software costs**
 - Access most software for free(-ish)
 - Google Docs suite vs Microsoft
 - Most applications are cloud-based
 - Rent (monthly payments) instead of buying

Advantages of Cloud Computing

- **Instant software updates**

- Avoid obsolete software and high upgrade costs
- Web-based apps update automatically
 - Available next login
 - Access latest version without paying or downloading
- Docker or VMs enable cross-platform software

- **Improved document format compatibility**

- Ensure document compatibility across users
- Eliminate format issues with shared cloud documents

Advantages of Cloud Computing

- **"Unlimited" storage capacity**
 - Cloud computing offers limitless storage
 - Your computer has ~1TB hard drive
 - Hundreds of PBs available in the cloud (elastic, on-demand)
- **Increased data reliability**
 - Hard disk can crash and destroy data
 - Few users regularly backup data
 - Cloud computer crash doesn't affect data storage
 - Data remains accessible in the cloud
 - Cloud computing is data-safe

Advantages of Cloud Computing

- **Universal document access**
 - Documents stay in the cloud
 - Instantly available wherever you are
 - Requires Internet connection
- **Latest version availability**
 - Edit at home, see changes at work
 - Cloud hosts latest version
 - Built-in revision control
- **Easier collaboration**
 - Sharing improves collaboration
 - Multiple users collaborate easily

Disadvantages of Cloud Computing

- Using cloud computing means dependence on Big Tech
 - AWS, Google, Microsoft monopolize the market
 - Limits flexibility and innovation
 - High barriers to enter the market
- Security is an issue
 - Unclear data safety
 - Unclear data ownership
- Issues relating to policy and access
 - Adhere to foreign policies if data stored abroad
 - Cloud providers store data within borders to comply with restrictions
 - E.g., TikTok saga
 - Remote server downtime affects file access
 - Users may be locked out and lose data access

Disadvantages of Cloud Computing

- **Requires a constant Internet connection**
 - Cannot access applications or documents without Internet
 - No Internet means no work
 - Some applications offer offline capabilities
- **Does not work well with low-speed/spotty connections**
 - Low-speed Internet makes cloud computing difficult
 - Web-based applications need high bandwidth
- **Web-interface can be slow**
 - Often slower than desktop even with fast connection
 - Interface sent between your computer and cloud
 - Latency is crucial

Disadvantages of Cloud Computing

- **Features might be limited**

- Web-based apps often lack features compared to desktop apps
- Microsoft PowerPoint offers more features than Google Slides
- Situation is changing rapidly

- **Stored data might not be secure**

- Cloud computing stores data on the cloud
- Is the cloud secure?
 - Likely more secure than your laptop
- Can unauthorized users access your data?
 - Data leaks are frequent

- **Stored data can be lost**

- Cloud data is safe, replicated across machines
- Local backups are still advisable

Disadvantages of Cloud Computing

- **HPC systems**

- Run compute-intensive HPC applications using MPI or OpenMP
- Require low-latency, high-bandwidth interconnect
- Schedule applications effectively
- Co-locate nodes to minimize communication latency
- Evolving landscape (e.g., MapReduce, AWS EC2)

- **Interoperability**

- Cloud systems use different protocols and APIs
- Running applications across different cloud systems may be challenging
- Solution: Use indirection layer (Terraform, Ansible)