Cloud-Based Data Processing

Data Centers

Jana Giceva, Michalis Georgoulakis

Datacenter Overview

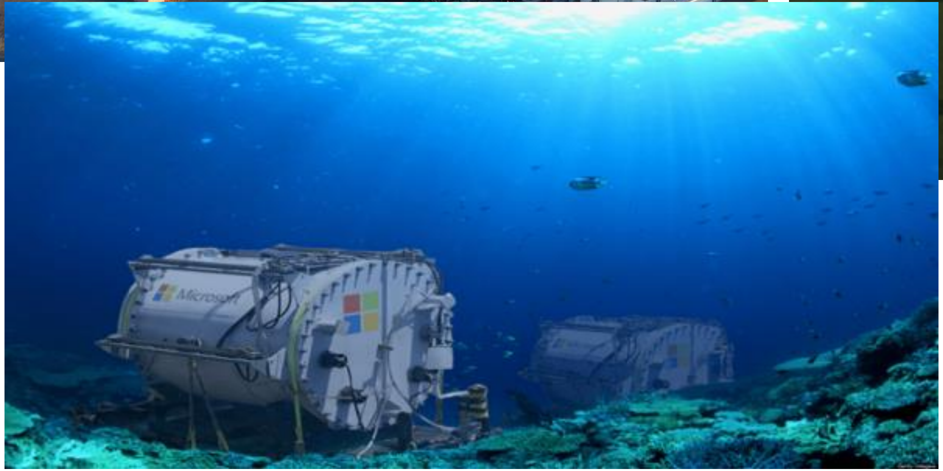
Data Centers

- Data center (DC): a physical facility that enterprises use to house computing and storage infrastructure in a variety of networked formats.

- Main function is to deliver utilities needed by the equipment and personnel:
  - Power
  - Cooling
  - Shelter
  - Security

- Size of typical data centers:
  - 500 – 10000 sq. m. buildings
  - 1 MW to 10-20 MW power (avg 5 MW)

Example data centers

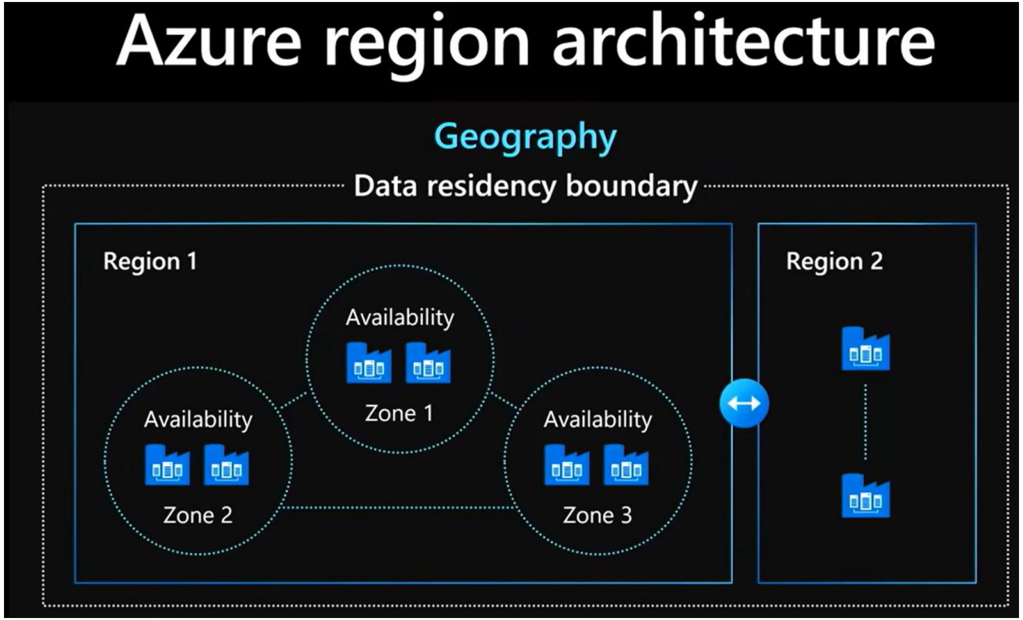


Datacenters around the globe

https://docs.microsoft.com/en-us/learn/modules/explore-azure-infrastructure/2-azure-datacenter-locations

Modern DC for the Cloud architecture

- **Geography:**
  - Two or more regions
  - Meets data residency requirements
  - Fault-tolerant from complete region failures
- **Region:**
  - Set of datacenters around metropolitan area
  - Usually within a ~100km continuous distance
  - Network latency perimeter < 2ms
- **Availability Zones:**
  - Unique physical locations within a region
  - Each zone made up of one or more DCs
  - Independent power, cooling, networking
  - Inter-AZ network latency < 1ms
  - Fault tolerance from DC failure



Src: Inside Azure Datacenter Architecture with Mark Russinovich

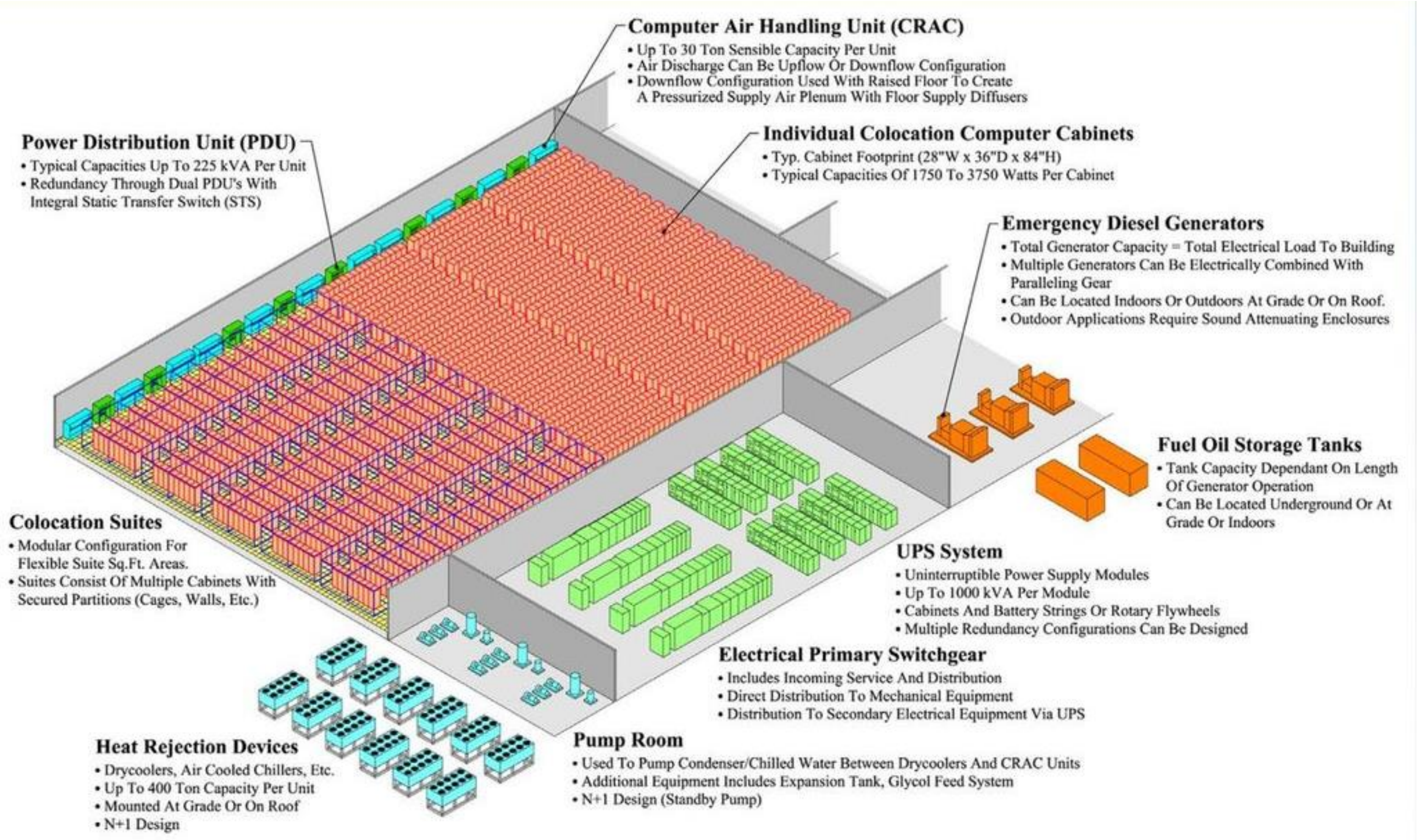
Datacenter Architecture

Data Centers vs Supercomputers

- **Scale-up**: high cost powerful CPUs, more cores, more memory
- **Scale-out**: adding more low cost, commodity servers

- Supercomputer vs. data center (numbers shown are rather old, take them with a grain of salt)
- **Scale**
  - Blue waters = 40K 8-core “servers”
  - Microsoft Chicago Data centers = 50 containers = 100K 8-core servers
- **Network architecture**
  - Supercomputers: InfiniBand, low-latency, high bandwidth protocols
  - Data Centers: (mostly) Ethernet based networks
- **Storage**
  - Supercomputers: separate data farm
  - Data Centers: use disk on node + memory cache

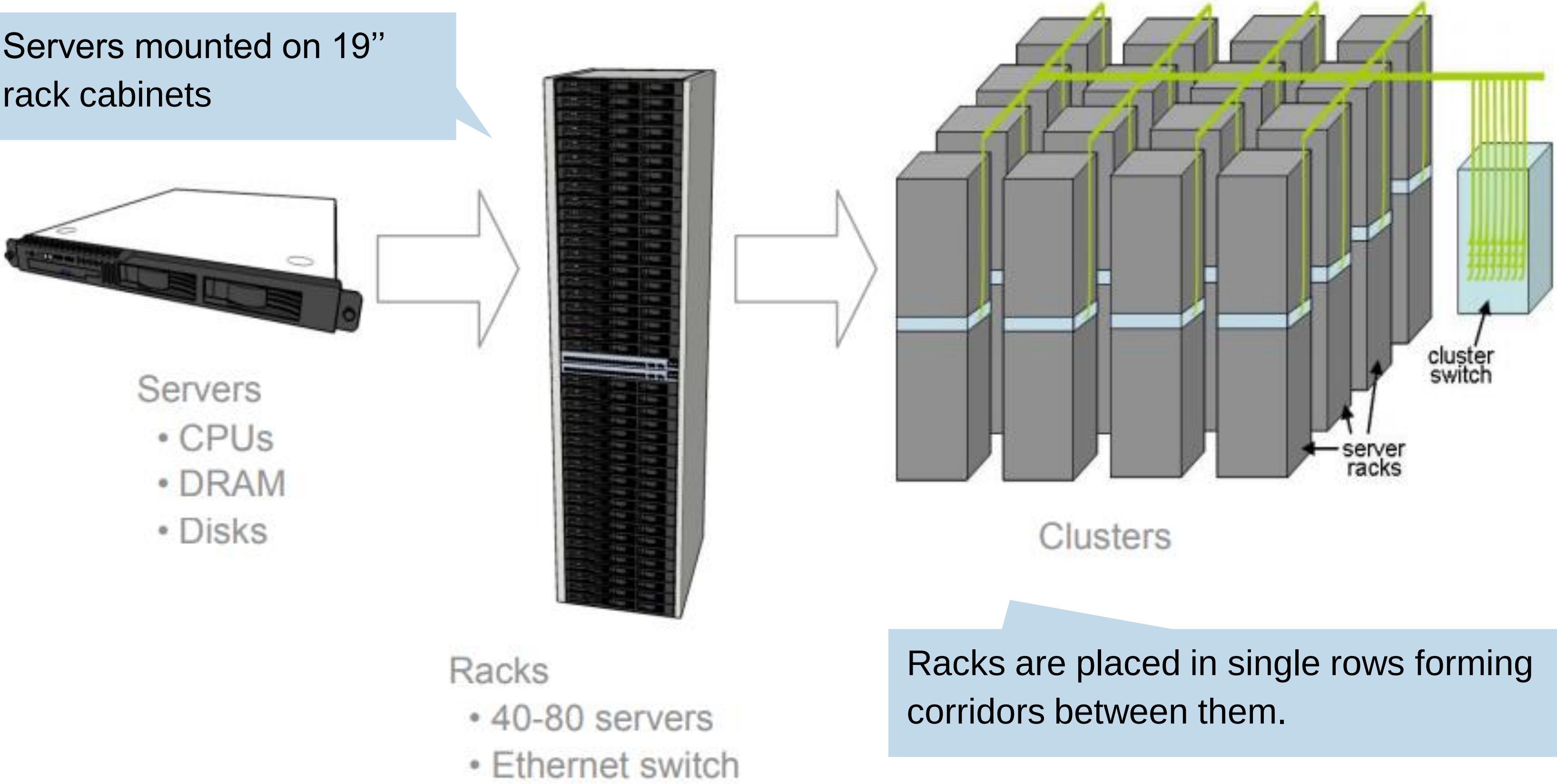
Main components of a datacenter



src: The Datacenter as a Computer – Barroso, Clidaras, Holzle

Traditional Data Center Architecture

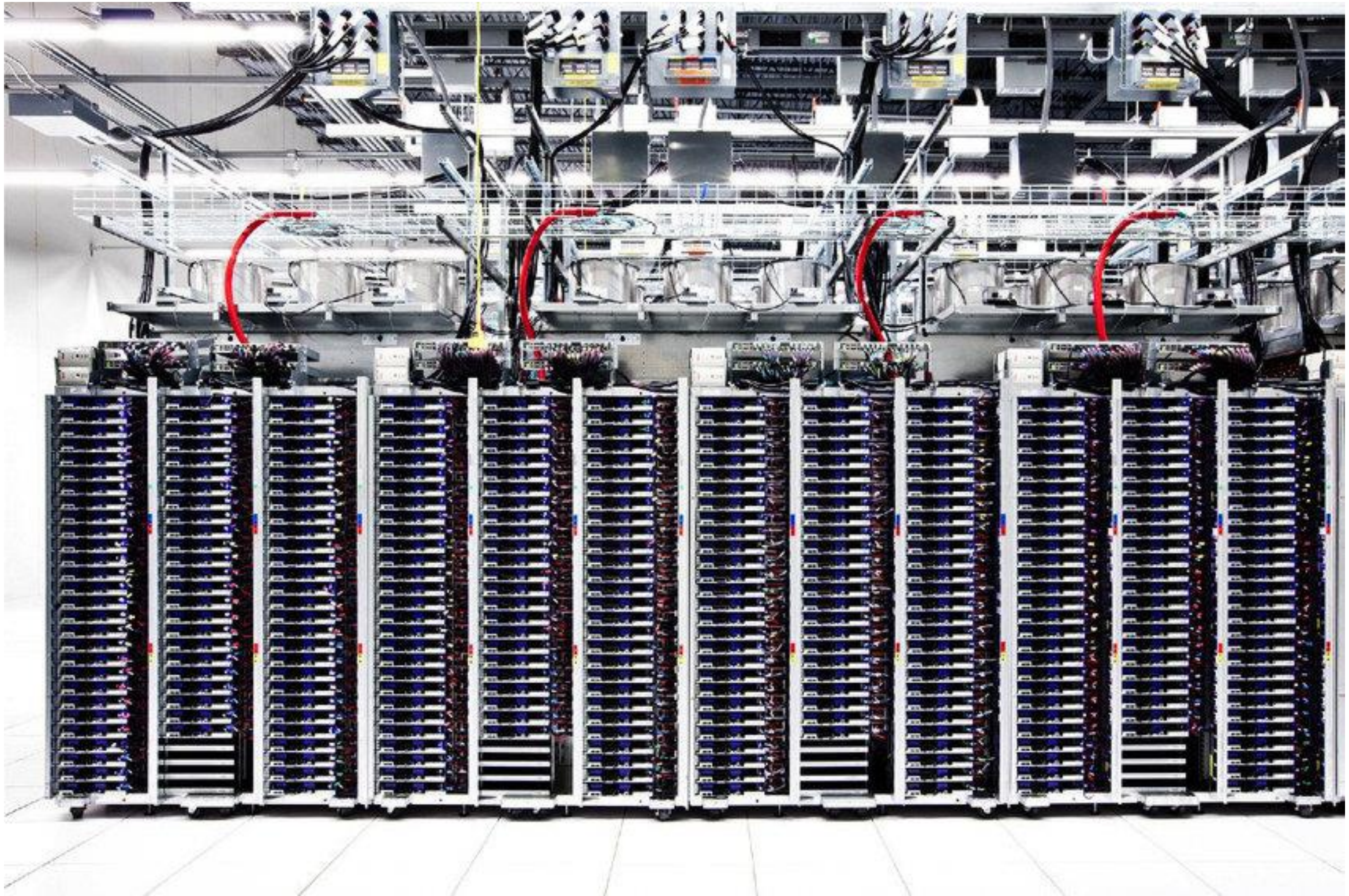
Servers mounted on 19'' rack cabinets



Racks are placed in single rows forming corridors between them.

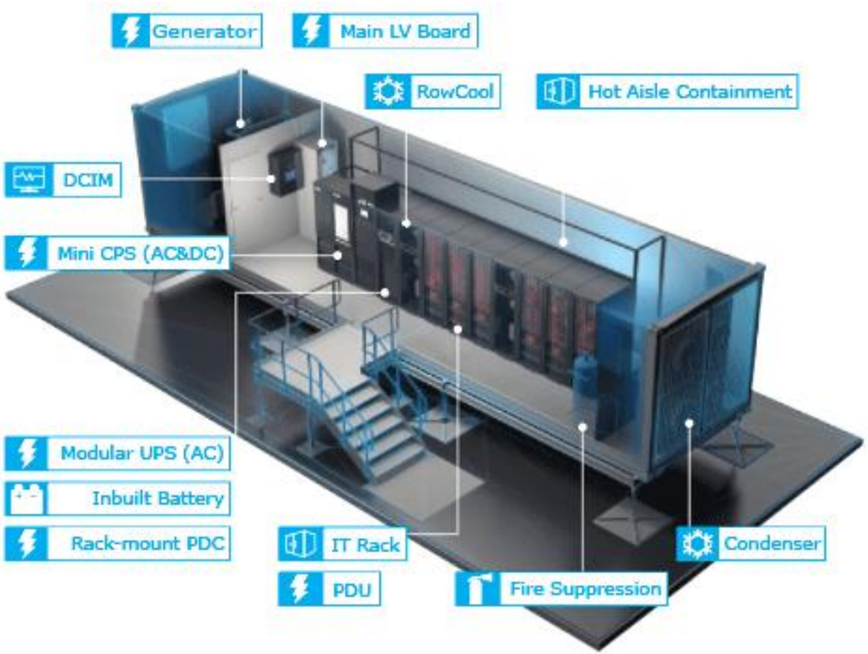
Src: the datacenter as a computer – an introduction to the design of warehouse-scale machines

A Row of Servers in a Google Data Center



Src: the datacenter as a computer – an introduction to the design of warehouse-scale machines

Inside a modern data center

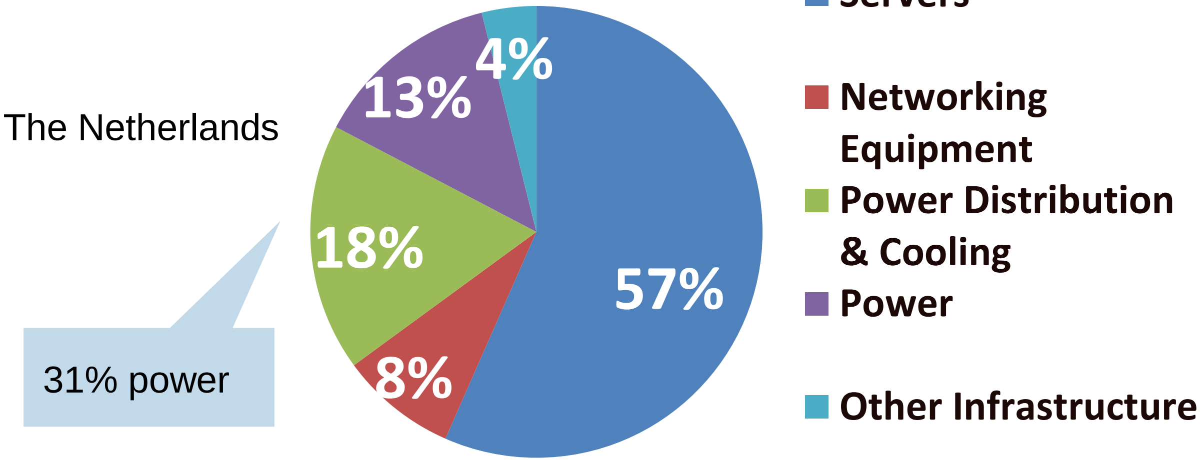


- Today's DC use shipping containers packed with 1000s servers each.
- For repairs, whole containers are replaced.

Costs for operating a data center

- DCs consume 3% of global electricity supply (416.2 TWh > UK's 300 TWh)
- DCs produce 2% of total greenhouse gas emissions
- DCs produce as much CO_2 as The Netherlands or Argentina

Monthly cost = $3'530'920



45,978 servers, 3yr server & 10 yr infrastructure amortization

Data Centers

- **Traditional data centers**
  - Host a large number of relatively small- or medium-sized applications, each running on a dedicated hardware infrastructure that is decoupled and protected from other systems in the same facility
  - Usually for multiple organizational units or companies
- **Modern data centers** (a.k.a., Warehouse-scale computers)
  - Usually belong to a single company to run a small number of large-scale applications
    - Google, Facebook, Microsoft, Amazon, Alibaba, etc.
  - Use a relatively homogeneous hardware and system software
  - Share a common systems management layer
  - Sizes can vary depending on needs

Power Usage Effectiveness (PUE)

- **PUE is the ratio of**
  - The total amount of energy used by a DC facility
  - To the energy delivered to the computing equipment

- **PUE is the inverse of data center infrastructure efficiency**

- **Total facility power** = covers **IT systems** (servers, network, storage) +
  other equipment (cooling, UPS, switch gear, generators, lights, fans, etc.)

PUE	Level of efficiency	DCIE
3.0	Very Inefficient	33%
2.5	Inefficient	40%
2.0	Average	50%
1.5	Efficient	67%
1.2	Very efficient	83%

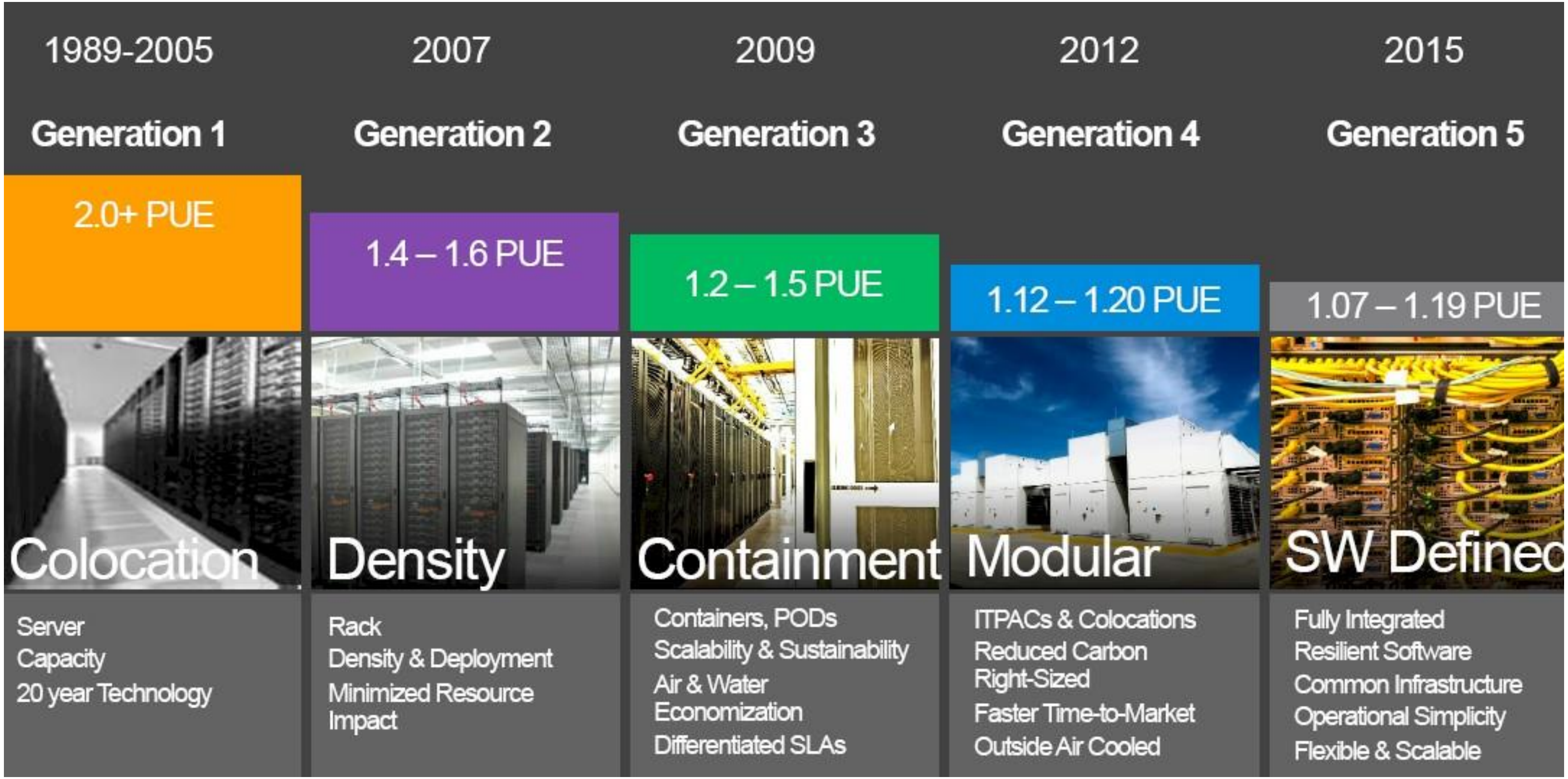
Source: https://www.42u.com/measurement/pue-dcie.htm

Achieving PUE

Price per Kilo Watt Hour	Where?	Possible Reason Why
3.6 cents	Idaho	Hydroelectric Power; Not Sent Long Distance
10.0 cents	California	Electricity Transmitted Long Distance over the Grid; Limited Transmission Lines in the Bay Area; No Coal Fired Electricity Allowed in California.
18.0 cents	Hawaii	Must Ship Fuel to Generate Electricity

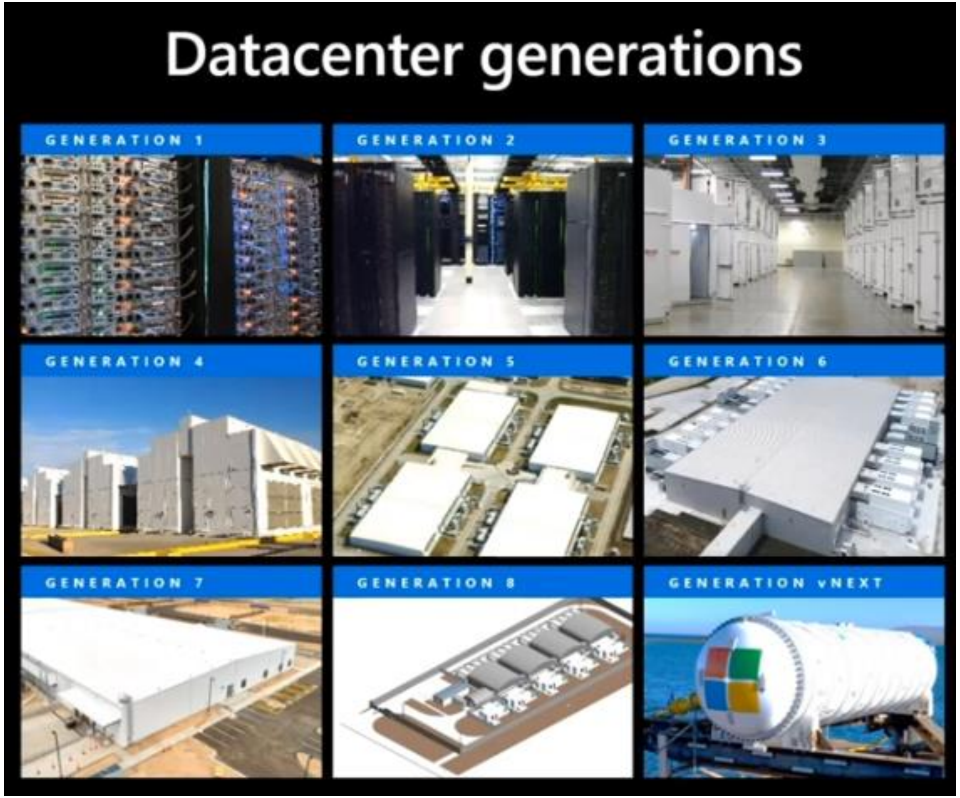
- **Location of the DC** – cooling and power load factor
- **Raise temperature of aisles**
  - Usually 18-20 C; Google at 27 C
  - Possibly up to 35 C (trade off failures vs. cooling costs)
- **Reduce conversion of energy**
  - E.g., Google motherboards work at 12V rather than 3.3/5V
- **Go to extreme environments**
  - Arctic circle (Facebook)
  - Floating boats (Google)
  - Underwater DC (Microsoft)
- **Reuse dissipated heat**

Evolution of data center design



https://www.nextplatform.com/2016/09/26/rare-tour-microsofts-hyperscale-datacenters/

Evolution of datacenter design

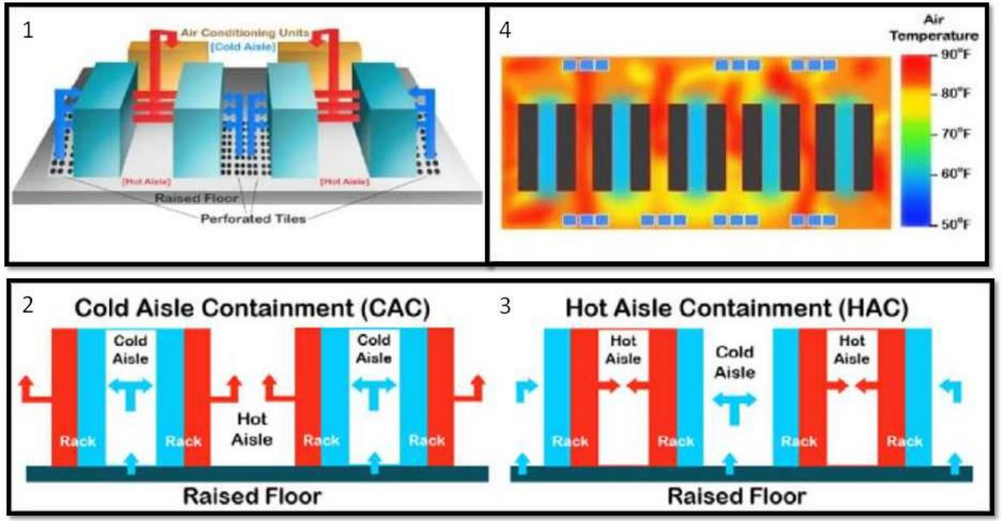


Src: Inside Azure Datacenter Architecture with Mark Russinovich

- **Gen 6: scalable form factor (2017)**
  - Reduced infrastructure, scale to demand
  - 1.17-1.19 PUE
- **Gen 7: Ballard (2018)**
  - Design execution efficiency
  - Flex capacity enabled
  - 1.15-1.18 PUE
- **Gen 8: Rapid deploy datacenter (2020)**
  - Modular construction and delivery
  - Equipment skidding and preassembly
  - Faster speed to market
- **Project Natick (future) – 1.07 PUE or less**

Datacenter Challenges

Challenge 1: Cooling data centers



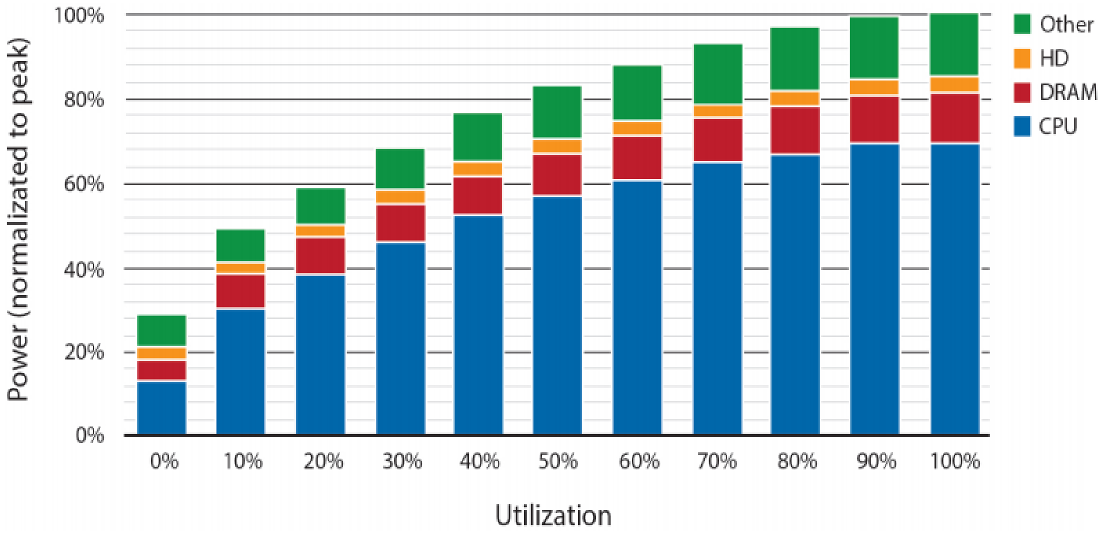
1- Conventional cooling
2- Cold Aisle Containment (CAC)
3- Hot Aisle Containment (HAC)
4- Thermal modelling

Cooling plant at a Google DC in Oregon

Challenge 2: Energy Proportional Computing

- **Average real world DC and servers are too inefficient.**
  – waste 2/3+ of their energy
- **Energy consumption is not proportional to the load**
  – CPUs are not so bad but the other components are
  – CPU is the dominant energy consumer in servers – using 2/3 of energy when active/idle.
- **Try to optimize workloads**
- **Virtualization and consolidation.**

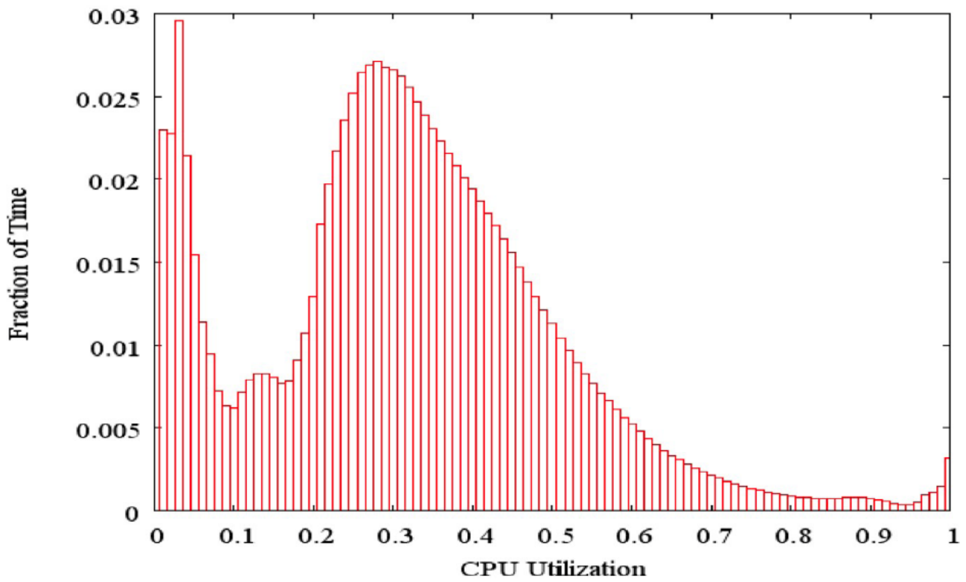
Sub-system power usage in an x86 server as the compute load varies from idle to full (reported in 2012).



src: “The Datacenter as a Warehouse Computer”

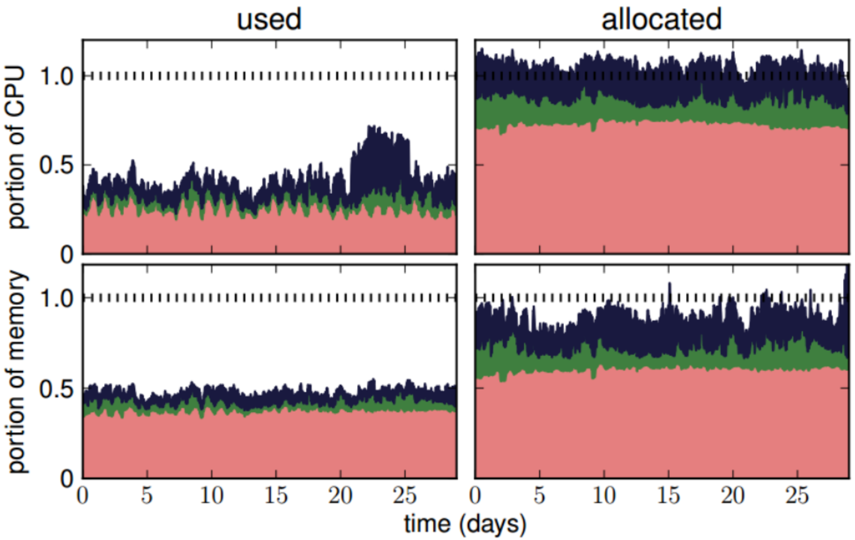
Challenge 3: Servers are idle most of the time

- For **non-virtualized servers 6-15% utilization**
- **Server virtualization** can **boost** to an average **30% utilization**
- **Need** for **resource pooling** and application and server **consolidation**
- Need for **resource virtualization**



src: Luiz Barroso, Urs Hölzle “The Datacenter as a Computer”

Challenge 4: Efficient monitoring



- Even with virtualization and software defined DC, **resource utilization can be poor**.
- Need for **efficient monitoring** (measurement) and **cluster management**.
- Goal to **meet SLOs** and **SLIs.**
- **Job's tail latency matters!**

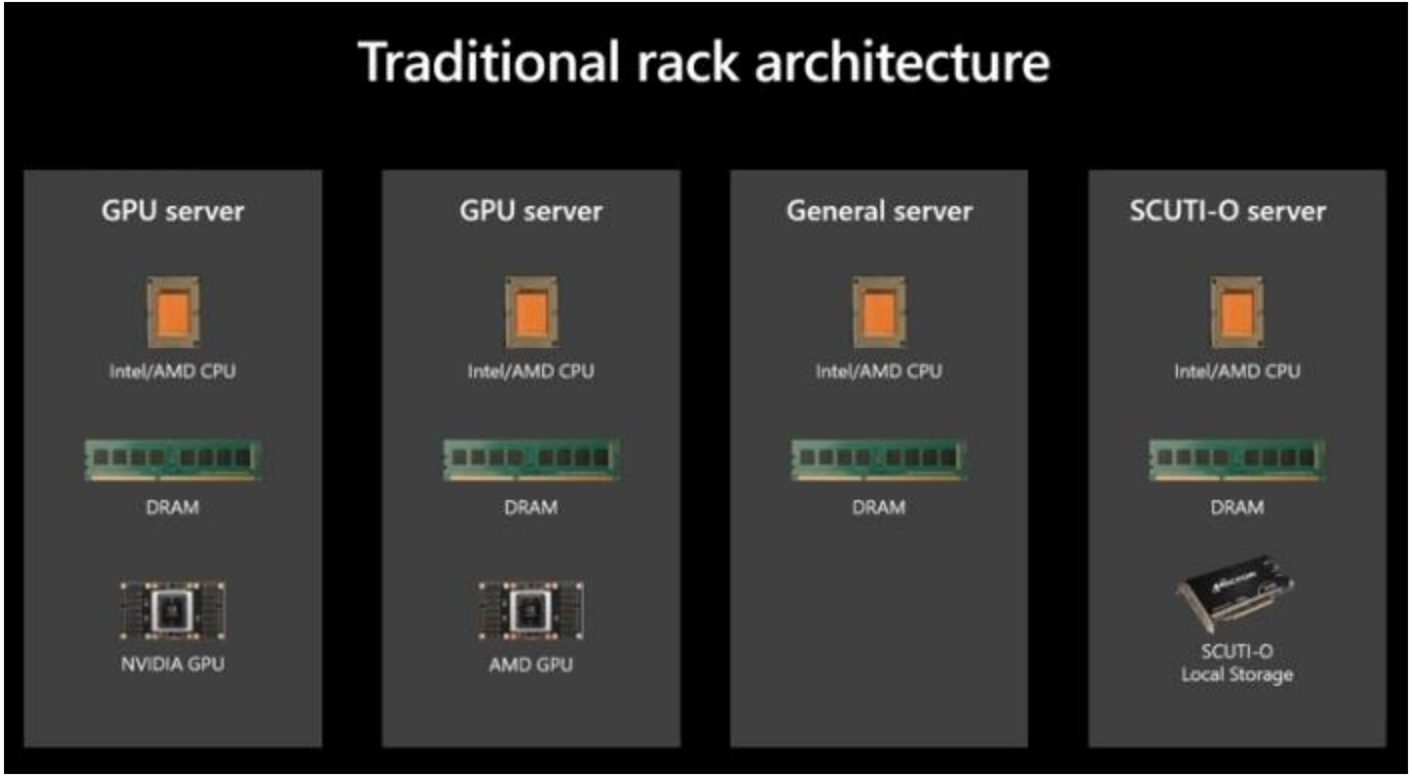
src: "Heterogeneity and dynamicity of clouds at scale: Google trace analysis" SoCC'12

Improving resource utilization

- **Hyper-scale system management software**
  - Treat the datacenter as a warehouse scale computer
  - Software defined datacenters
  - System software that allows DC operations to manage the entire DC infrastructure
  - Compose a system using pooled resources of compute, network, and storage based on workload requirement

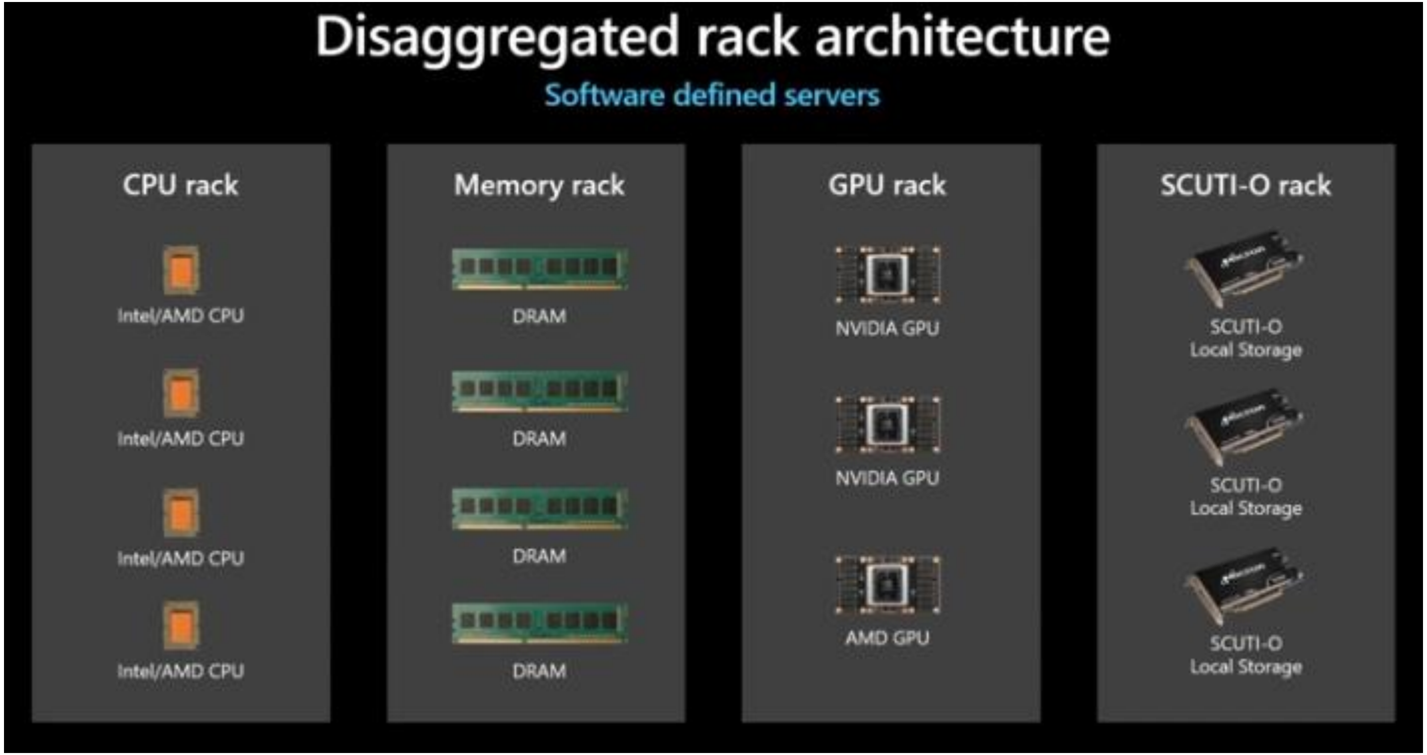
- **Dynamic resource allocation**
  - Virtualization is not enough to improve efficiency
  - Need the ability to dynamically allocate CPU resources across servers and racks, allowing admins to quickly migrate resources to address the shifting demand
  - Drive 100-300% better utilization for virtualized WLs, and 200-600% for bare-metal WLs.

Software defined DCs



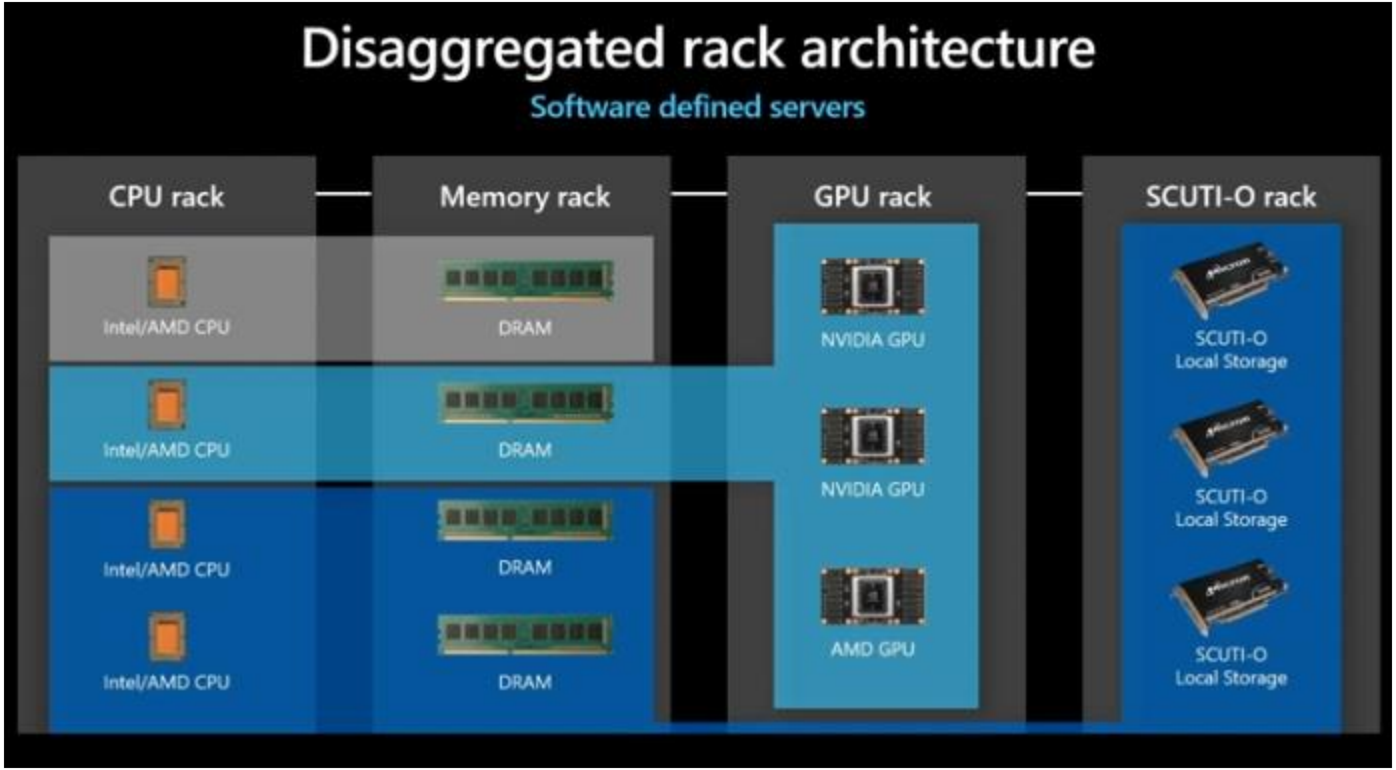
Src: Inside Azure Datacenter Architecture with Mark Russinovich

Disaggregation across racks



Src: Inside Azure Datacenter Architecture with Mark Russinovich

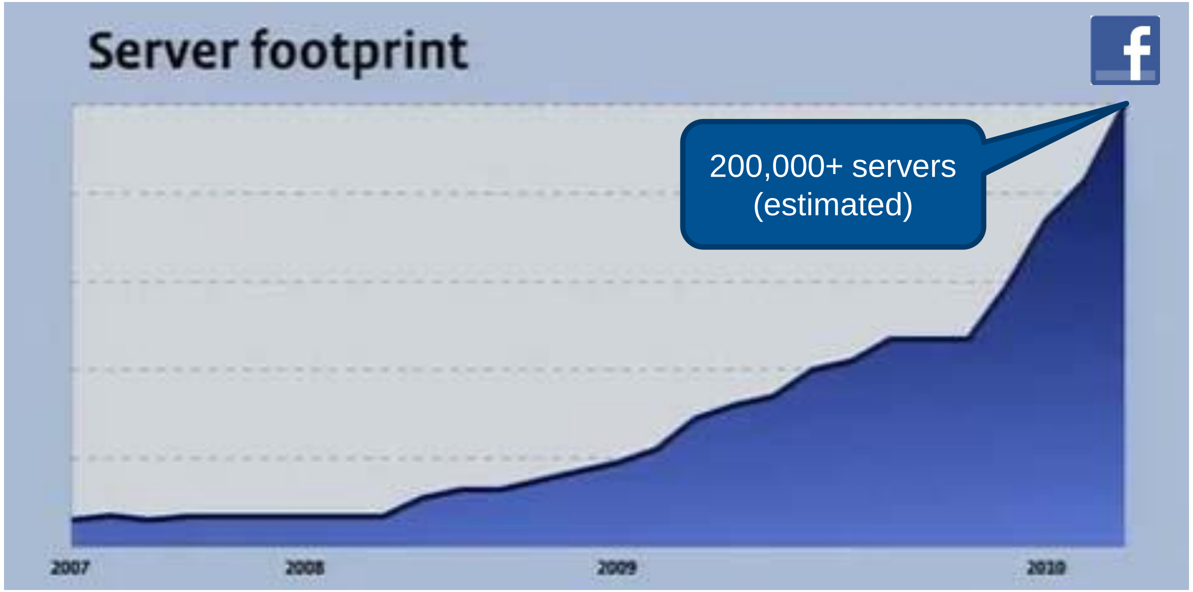
Software defined Servers



Src: Inside Azure Datacenter Architecture with Mark Russinovich

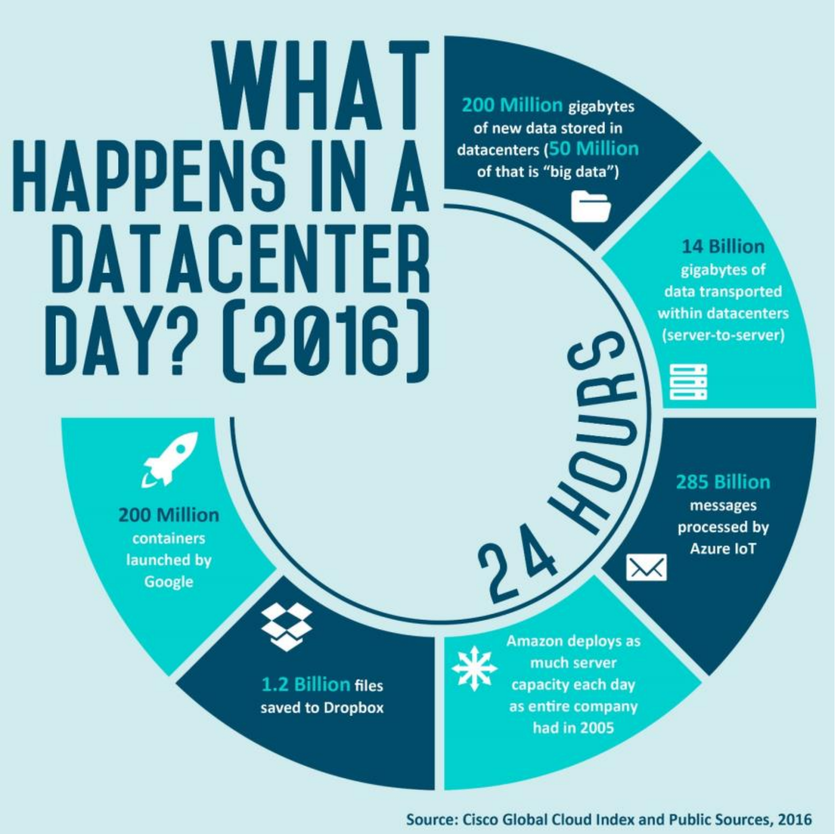
Challenge 5: Managing scale and growth

- In 2016, Gartner estimated that Google has 2.5 million servers
- In 2021, Microsoft Azure was reported to have more than 4 million servers in operations globally.
- All big 3 vendors are estimated to have more than 5 millions today.

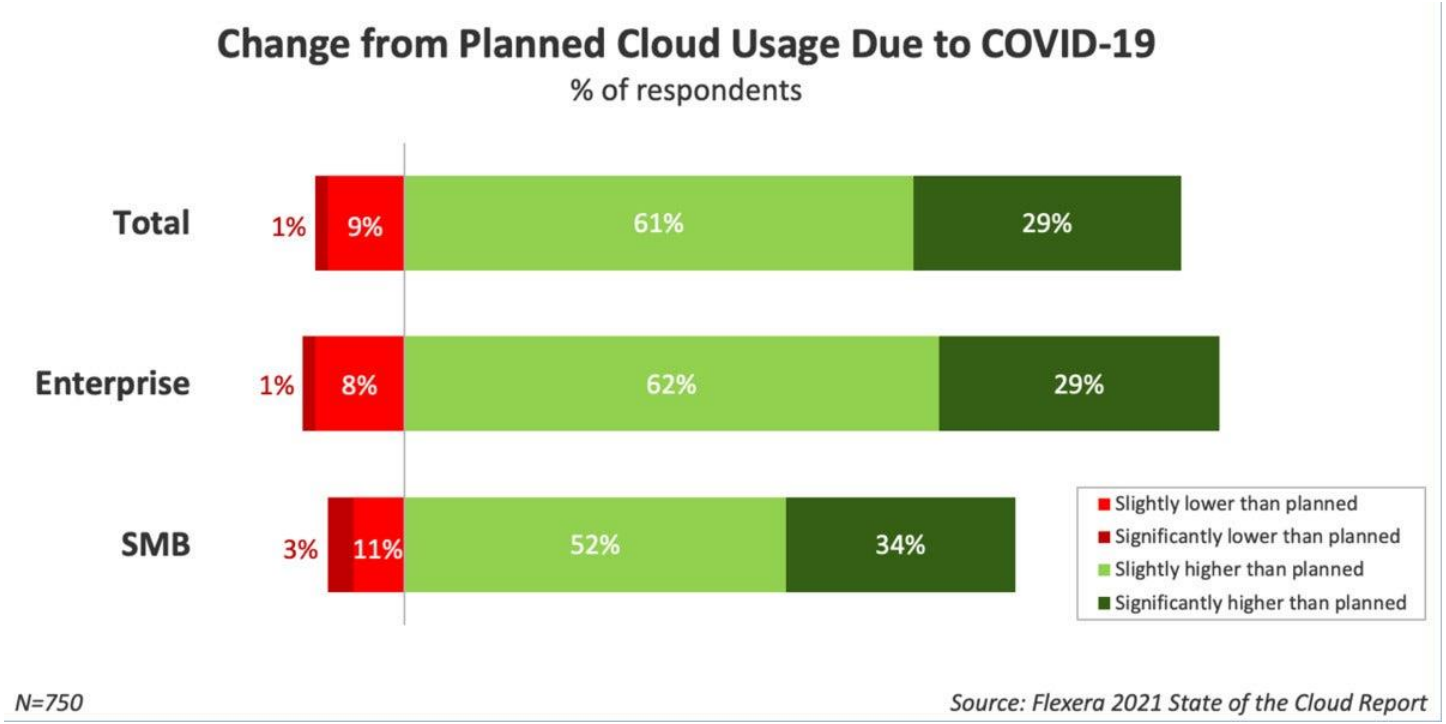


Size and growth of DC (2016-2020)

- The scale and complexity of DC operations grows constantly.
- By 2020, Cisco estimated that we would have 600 million GB of new data saved each day (200 million GB big data)
- So the volume of BigData by 2020 was estimated to be as much as all of the stored data in 2016.

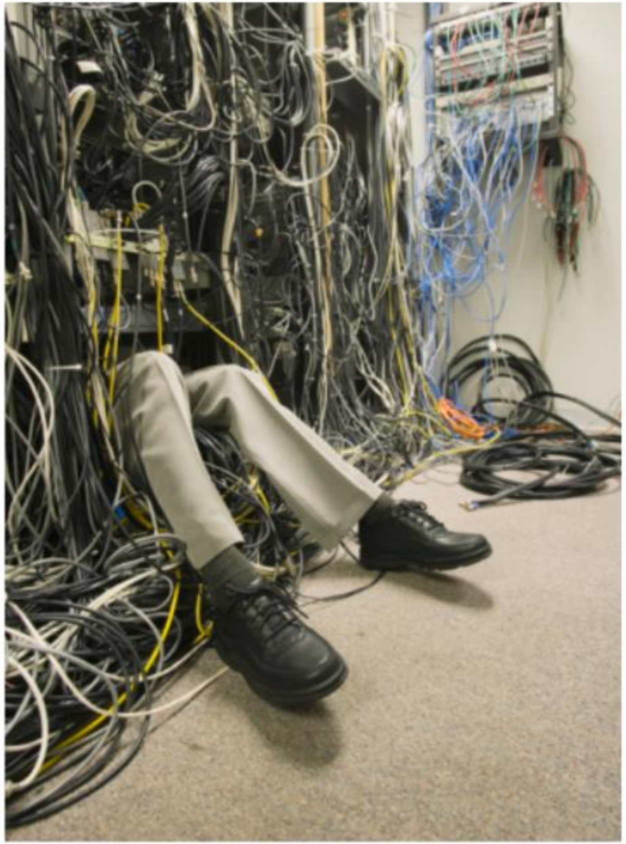


Impact of COVID Pandemic

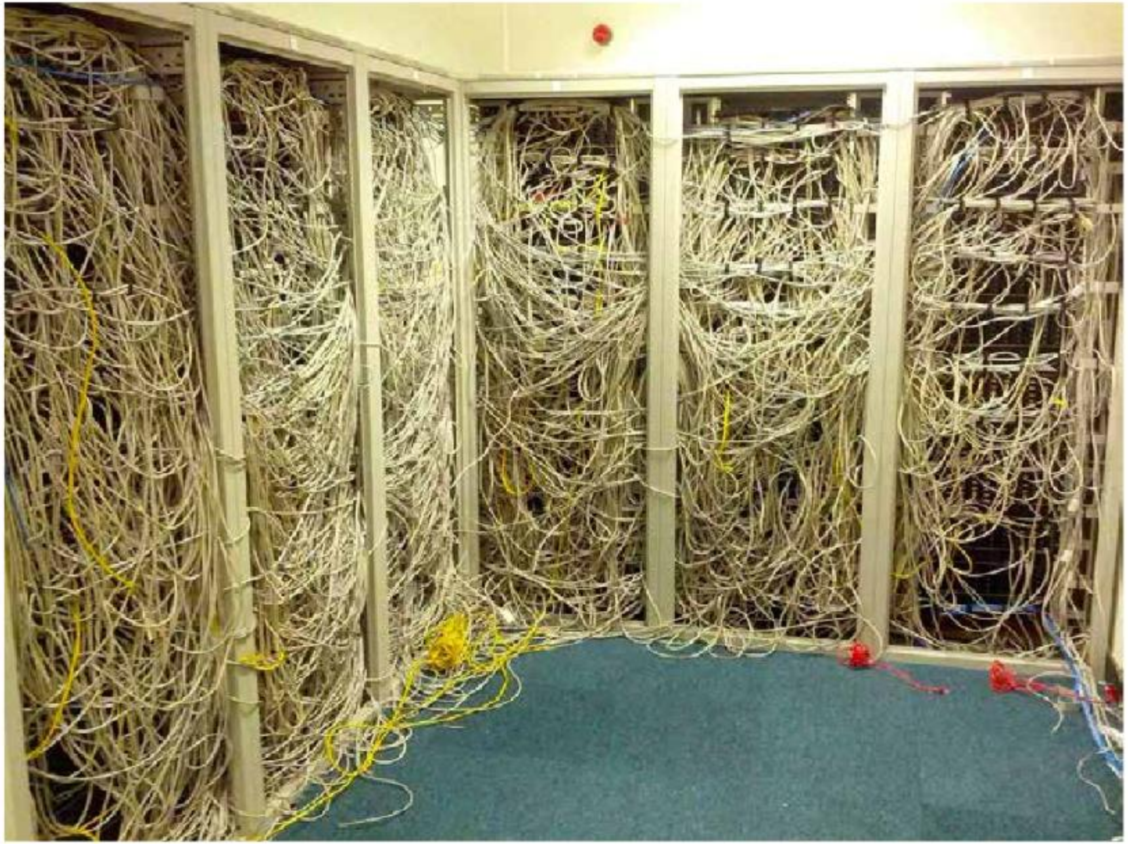


- 9 out of 10 companies accelerated their cloud adoption in response to the pandemic

Challenge 6: networking at scale



[David Samuel Robbins, gettyimages.ch]



[@AlexCWheeler, Twitter]

Challenge 6: networking at scale (cont.)

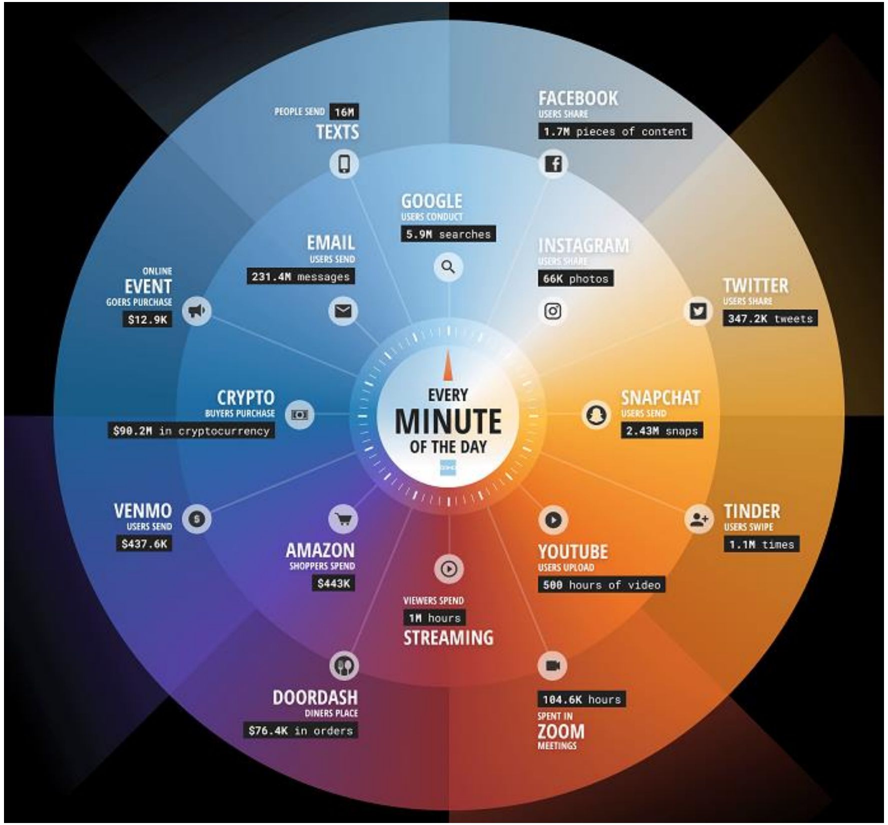
- **Building the right abstractions to work for a range c**

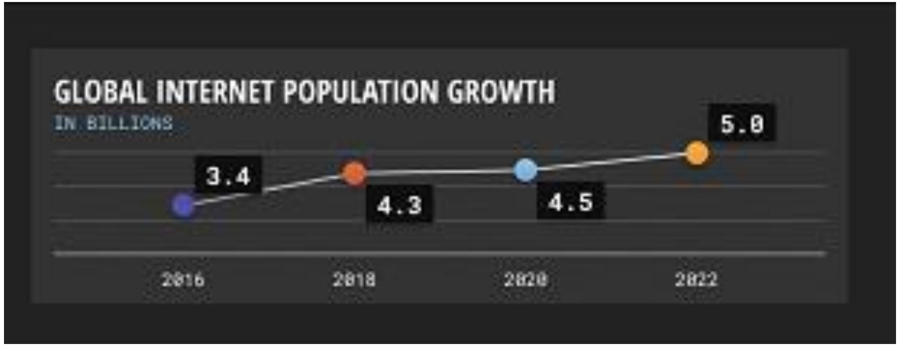
Software Defined Networkin

- **Within DC, 32 billion GBs will be transported in 202**
  - src: Cisco's report 2016-2020
- **"Machine to machine" traffic is orders of magnitude**
  - Src: Jupiter Rising: A Decade of Clos Topologies and network (ACM SIGCOMM'15)
- **Evolution via optical circuit switches and SDN**
  - Src: Jupiter Evolving: Transforming Google's Datacer (ACM SIGCOMM'22)

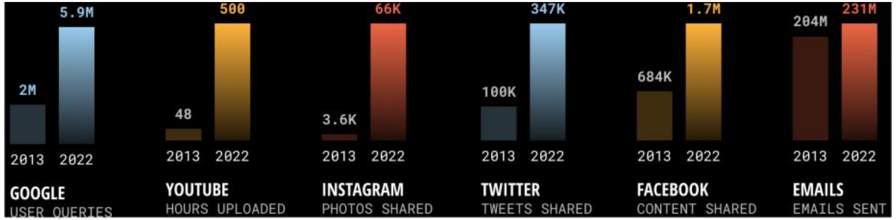
Cloud Computing Overview

Big Data and the need for Cloud





- Over the last 10 years digital engagement, streaming content, online purchasing, p2p payments, etc. have risen by orders of magnitude.
- Src: https://www.domo.com/data-never-sleeps



Cloud and Cloud computing

Datacenter hardware and software that the vendors use to offer the computing resources and services.

- The cloud has a large pool of easily usable **virtualized** computing **resources**, development **platforms,** and various services and **applications.**

- Cloud computing is the delivery of **computing as a service.**

- The **shared resources, software,** and **data** are provided by a provider as a **metered service over a network.**

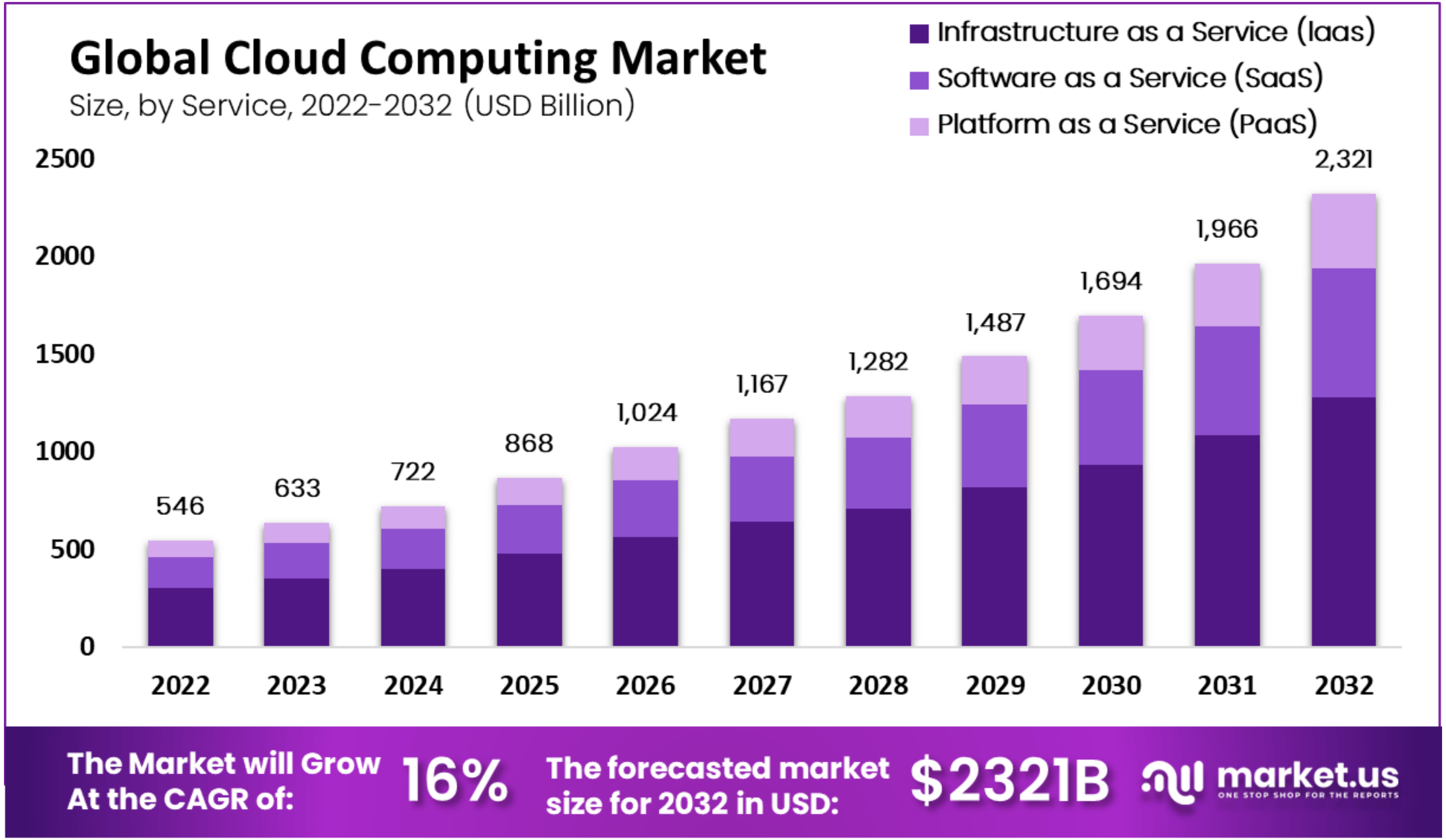
Cloud Computing

- **Datacenters are vendors that rent servers or other computing resources** (e.g., storage)
  - Anyone (or company) with a “credit card” can rent
  - Cloud resources owned and operated by a third-party (cloud provider).

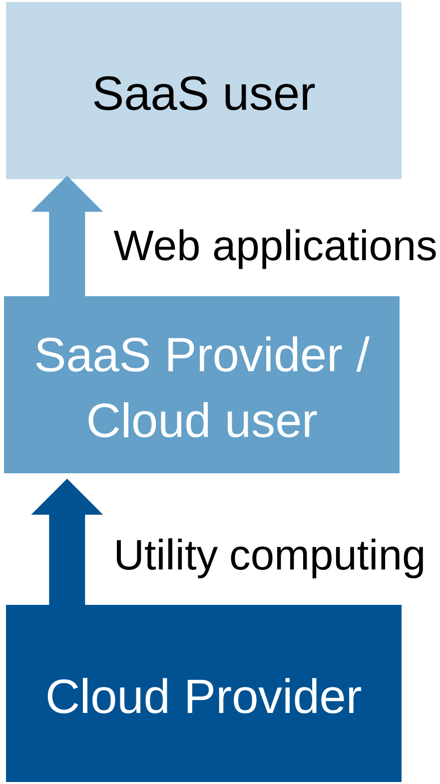
- **Fine-grained pricing model**
  - Rent resources by the hour (or by the minute) or by I/O
  - Pay as you go (pay for only what you use)

- **Can vary capacity as needed**
  - No need to build you own IT infrastructure for peak loads
  - Can reserve fixed pools of servers ahead of time, rent them as needed, or a combination of both.

Cloud market revenue in billions of dollars



Cloud Roles



Application users

Cloud Users

- **Software / websites that serve real users**
  - Netflix, Pinterest, Instagram, Spotify, AirBnB, Lyft, Slack, Expedia
- **Data analytics, machine learning, and other data services**
  - Databricks, Snowflake, GE Healthcare
- **Mobile and IoT backends**
  - Snapchat, Zynga (AWS → zCloud → AWS)
- **Datacenter's own software**
  - Google Drive/One Drive, search, etc.

Cloud Providers

- **Companies with large DCs**
  - Amazon AWS, Microsoft Azure, GCP, Alibaba Cloud, Oracle cloud

Types of Cloud Computing

Public vs. Private

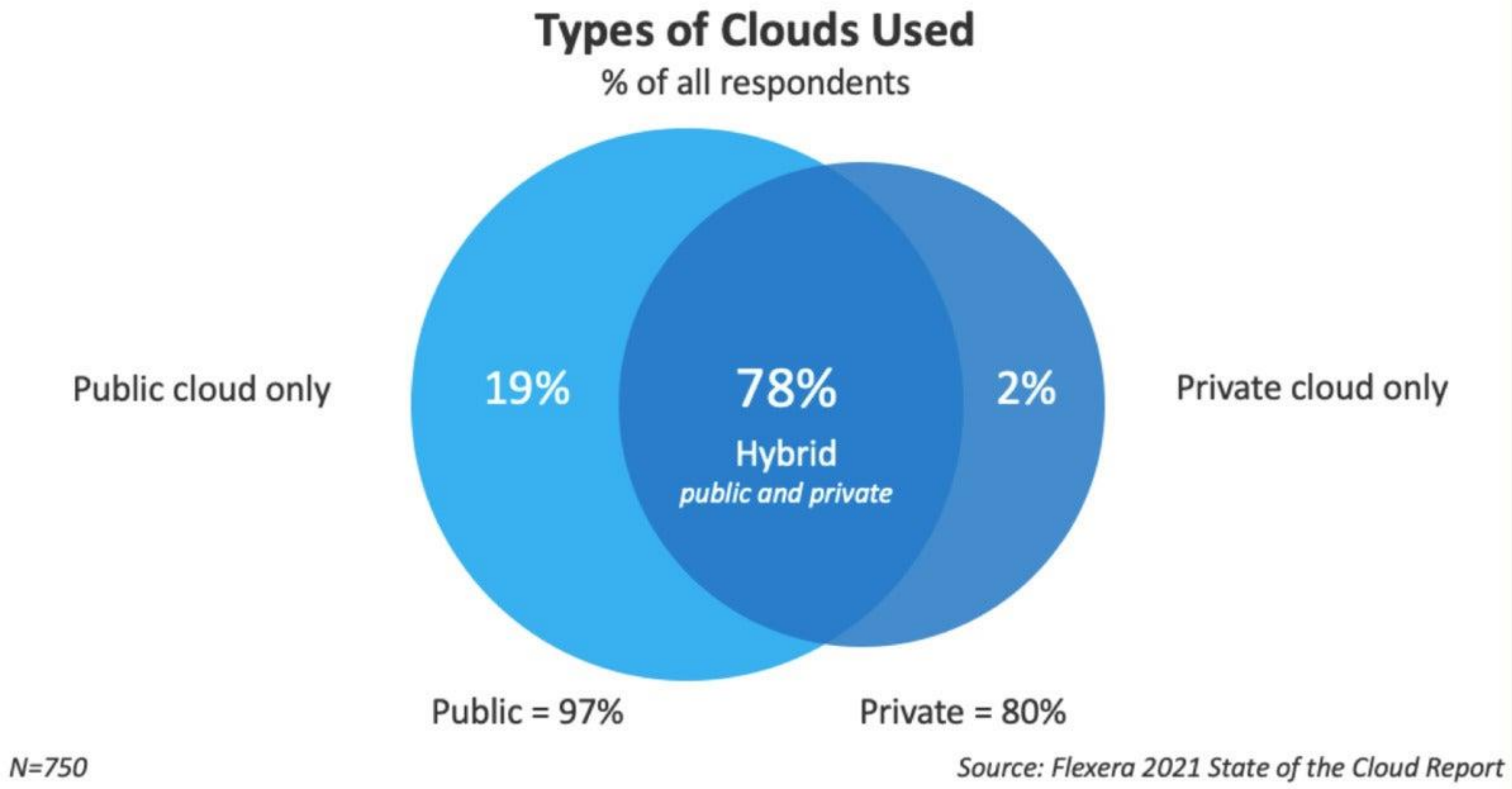
- Public: resources owned and operated by the one organization aka the cloud vendor
- Private: Resources used exclusively by a single business or organization

On-premise vs. Hosted:

- On-premise (on-prem): resources located locally (at a datacenter that the organization operates)
- Hosted: resources hosted and managed by a third-party provider

Private cloud can be both on-prem and hosted (virtual private cloud)

Types of Clouds Used



Types of Cloud Computing (cont)

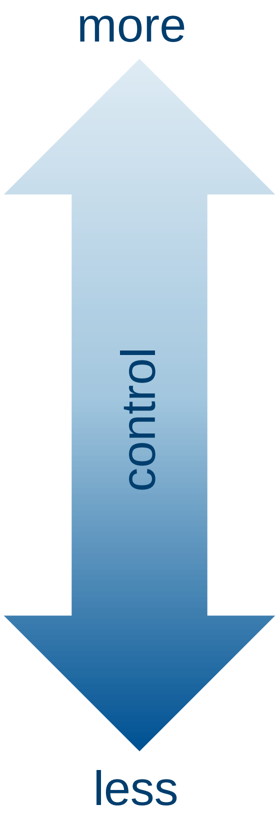
Hybrid cloud

- Combines public and private clouds, allows data and applications to be shared between them.
- Better control over sensitive data and functionalities
- Cost effective, scales well and is more flexible

Multi-Cloud

- Use multiple clouds for an application / service
- Avoids data lock-in
- Avoids single point of failure
- But, need to deal with API differences and handle migration across clouds

Cloud service models (XaaS)



Infrastructure as a Service (IaaS)

- Rent IT infrastructure – servers and virtual machines (VMs), storage, network, firewall, and security

Platform as a Service (PaaS)

- Get on-demand environment for development, testing and management of software applications: servers, storage, network, OS, databases, etc.

Serverless, Function as a Service (FaaS)

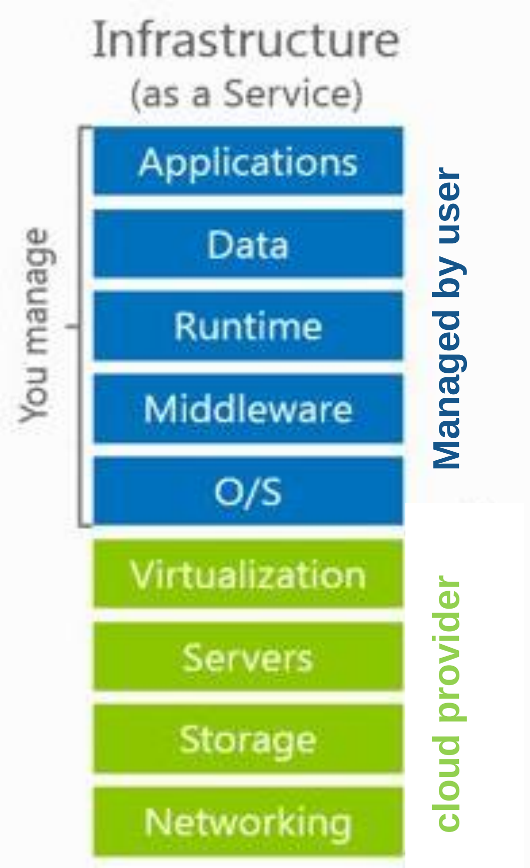
- Overlapping with PaaS, serverless focuses on building app functionality without managing the servers and infrastructure required to do so.
- Cloud vendors provides set-up, capacity planning, and server management.

Software as a Service (SaaS)

- Deliver software applications over the Internet, on demand.
- Cloud vendor handles software application and underlying infrastructure

Infrastructure as a Service

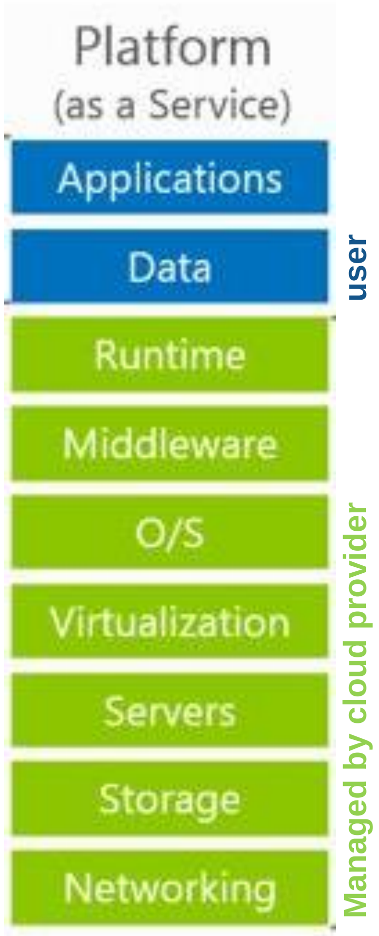
- Immediately **available computing infrastructure,** provisioned and managed by a cloud provider.
- Computing **resources pooled** together to server multiple users / tenants.
- Computing resources include:
storage, processing, memory, network bandwidth, etc.



src image from Microsoft Azure

Platform as a Service

- Complete development and deployment environment.
- Includes system's software (OS, middleware), platforms, DBMSs, BI services, and libraries to assist in development and deployment of cloud-based applications.
- Examples:
- What is serverless computing then?



src image from Microsoft Azure

Software as a Service

Cloud pros and cons

User’s benefits:

- Elimination of up-front commitment
- Speed – services are provided on demand
- Global scale and elasticity
- Productivity
- Performance and security
- Customizability
- Ability to pay for use of computing resources on a short-term basis (as needed)

User’s concerns:

- Dependability on network and internet connectivity
- Security and privacy
- Cost of migration
- Cost and risk of vendor lock-in

References

In addition to the cross references provided in the slides.

Some material based on:

- Lecture notes from “Scalable Systems for the Cloud” by Prof. Giceva at Imperial
- Lecture notes from “Modern Data Center Systems” by Prof. Zhang at UC San Diego

- Book “The Datacenter as a Computer – An Introduction to the Design of Warehouse-scale Machines” by Luiz Andre Barroso, Jimmy Clidaras, Urs Holzle
- Talk “Inside Azure Datacenter Architecture” with Mark Russinovich (Azure CTO)
- Paper “Above the Clouds: A Berkeley View of Cloud Computing”
- Web-pages from Amazon AWS, Microsoft Azure and Google CDP

Thank you for your attention!