Advanced Cloud Computing

Introduction & Logistics

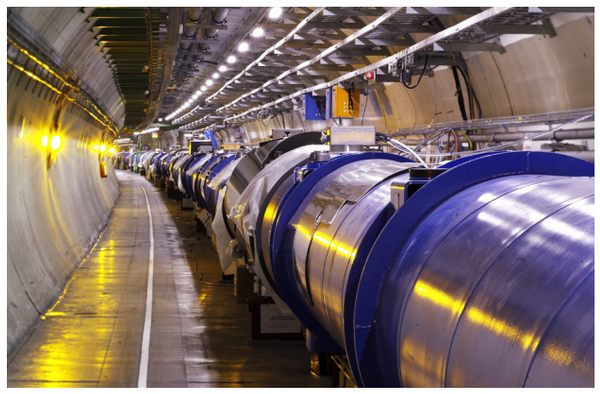
Wei Wang
CSE@HKUST
Spring 2022

THE DEPARTMENT OF
COMPUTER SCIENCE & ENGINEERING
計算機科學及工程學系

About me

- **Wei Wang**, Associate Professor, CSE
  - Email: weiwa@cse.ust.hk
  - Office: Rm3524
- Research interests
  - Distributed systems, with particular focus on cloud computing, big data and machine learning systems

Data, data, data!



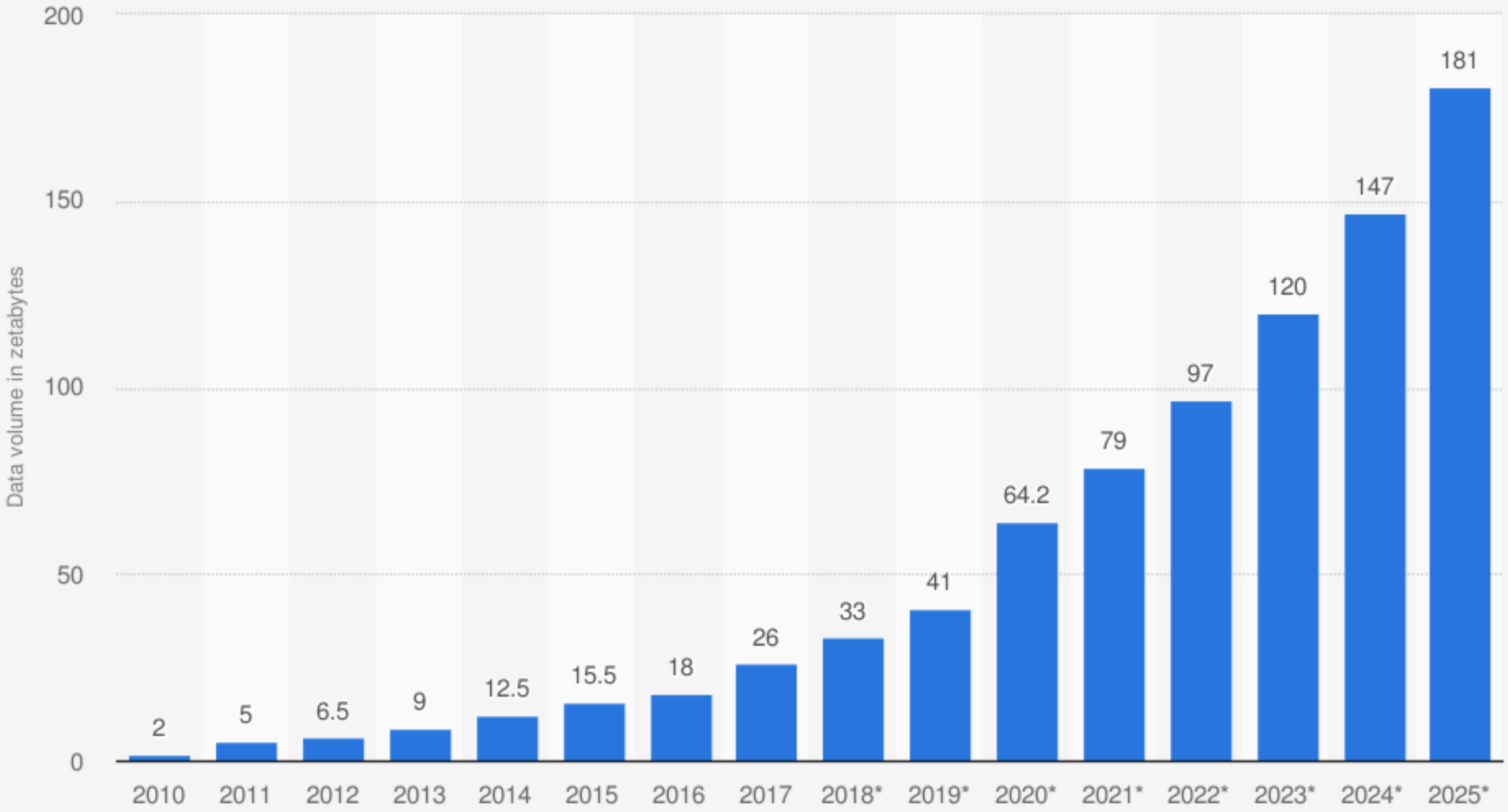
Large Hadron Collider generates 40 TB data per second

Boeing Jet Engine creates 10 TB operation information every 30 minutes

Google

Search index contains 100s billions ($>10^{11}$) webpages and is well over 100 petabytes ($>10^{17}$) in size

Volume of data/information created, captured, copied, and consumed worldwide from 2010 to 2025 (in zettabytes)



Sources
IDC; Seagate; Statista estimates

Additional Information:
Worldwide; 2010 to 2020

“640K ought to be enough for anybody.”
— Bill Gates (1981)

How can we crunch the massive amount of data?

Cloud Datacenter

Datacenters

- >100K servers
- Costs in billions of dollars
- Geographically distributed

It is estimated that >94% global workloads was processed in datacenters in 2021

“I think there is a world market for maybe five computers.”

— Thomas Watson (1943)

Cloud Computing

- **Computing as a utility:** deliver computing resources over the Internet, as a metered service
  - *Dynamic provisioning*: pay-as-you-go
  - *Scalability*: “infinite” capacity
  - *Elasticity*: scale up or down

amazon web services™ | EC2

	vCPU	ECU	Memory (GiB)	Instance Storage (GB)	Linux/UNIX Usage
General Purpose - Current Generation					
t2.micro	1	Variable	1	EBS Only	$0.013 per Hour
t2.small	1	Variable	2	EBS Only	$0.026 per Hour
t2.medium	2	Variable	4	EBS Only	$0.052 per Hour
t2.large	2	Variable	8	EBS Only	$0.104 per Hour
m4.large	2	6.5	8	EBS Only	$0.126 per Hour
m4.xlarge	4	13	16	EBS Only	$0.252 per Hour
m4.2xlarge	8	26	32	EBS Only	$0.504 per Hour
m4.4xlarge	16	53.5	64	EBS Only	$1.008 per Hour
m4.10xlarge	40	124.5	160	EBS Only	$2.52 per Hour
m3.medium	1	3	3.75	1 x 4 SSD	$0.067 per Hour
m3.large	2	6.5	7.5	1 x 32 SSD	$0.133 per Hour
m3.xlarge	4	13	15	2 x 40 SSD	$0.266 per Hour
m3.2xlarge	8	26	30	2 x 80 SSD	$0.532 per Hour

Now that we have computing resources in cloud. What's next?

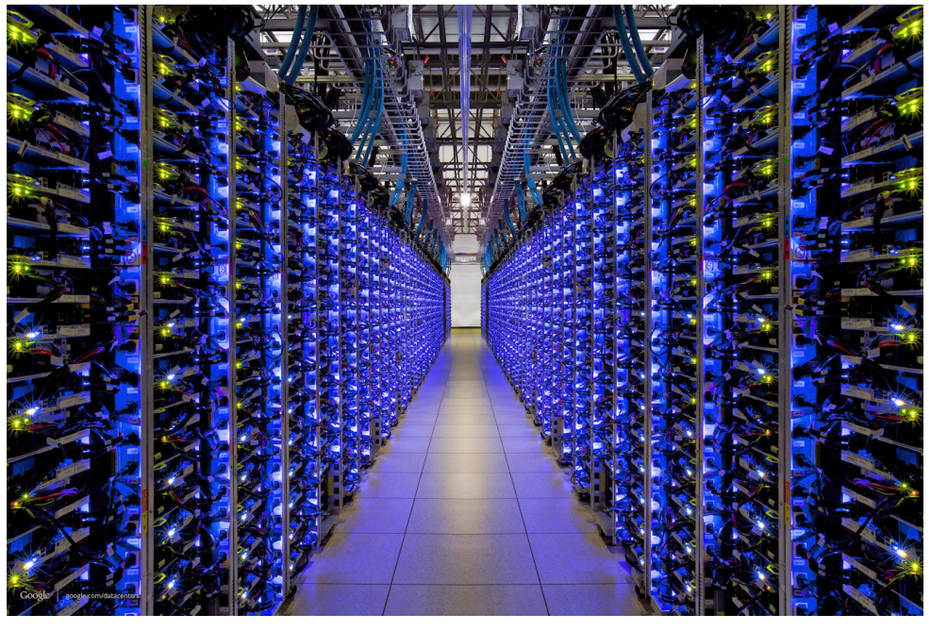
OS for the cloud: Cluster management systems & distributed computing frameworks

The datacenter **is** a computer

Windows

OS X Yosemite

kubernetes



About the course

- Website: in Canvas, CSIT6000O
- Announcements and course materials are posted online on a regular basis
- TAs:
  - Lingyun Yang (lyangbk@cse.ust.hk)

Prerequisites

- Object-oriented programming
- Data structures and algorithms
- Comfortable with Python/Java programming
- Comfortable with Unix/Linux
- A laptop that can host a Linux VM with at least 2 cores and 8 GB RAM (or a Mac/Linux laptop)
  - has **`VirtualBox`** installed
  - has **`Git`** installed

Accept an email invitation from AWS Academy Learner Lab to get $100 USD credit

A note on lab environment

- AWS Academy Learner Lab
  - Offers a long-running lab environment for students to learn cloud and AWS production services
    - Each student will have a **$100** AWS Platform Credit to spare
    - Have access to a **restricted set of AWS services**
      - we will mainly use EC2, S3, EMR, and Lambda
- Extra expense at your own cost. **We (me, or the dept) cannot financially help you in any means!**

Textbook/References

- No official textbook
  - Cloud computing is a rapidly evolving technology
- Best way to learn it is to read research papers
  - Landmark and cutting-edge research work that developed the cloud technology
  - e.g., Google Filesystem, MapReduce, Spark
  - Reading list will be posted online

Learning by reading & doing,
and learn things online!

Assessment

- Homework and labs (30%)
  - programming on AWS and locally hosted VMs
- Midterm exam (20%)
- Open-ended course project (50%)
- No final

Course project

- Teamwork
  - a group of 2-4 students
- Open-ended (sample topics available as well)
  - Must be related to cloud computing
  - Engineering/research
- Project report (30%) + presentation (20%)

Course project

- Example topics
  - Data analytics on public datasets (e.g., AWS Public Datasets)
  - A cloud-based service application (e.g., reimplementing MapReduce/Spark framework using AWS Lambda)
  - Cloud resource management and scheduling using Kubernetes
  - Reproducing a published cloud system work
  - …

To pass

- Attend the lecture and tutorial
- Get your hands dirty
- Learn things online
- Do all assignments by yourself
- Do well in the exam

You cannot have a good understanding of Cloud without trying it yourself

Academic honesty

- In short, don’t cheat!
- Don’t copy code or solutions from your classmates or third-party sources, and don’t let others copy yours. Both cases are plagiarism and penalized in the same way

Protocol for Plagiarism

- We will detect possible plagiarism in your code/reports.
- Suspicious cases will be directly reported to the CS general office. A panel will be formed to deal with all cases.
- Minimum penalty: zero mark for the assignment/ homework.

Other matters

- Try to come on time
- Participate as much as you can in the classroom. It’s a two-way avenue.

S. Keshav, “How to Read a Paper,” ACM SIGCOMM Comput. Comm. Rev. 2007

The three-pass approach

- **The first pass** (5 - 10 min): get the general idea of the paper

- If needed, go to **the second pass** (1 hour): grasp the paper's content, but not details

- If needed, go to **the third pass** (several hours or days): *virtually re-implement* the ideas and technical details

The first pass is to get a bird's eye-view of the paper (5 - 10 min)

The first pass

- Carefully read the title, abstract and introduction
- Only read the section and sub-section headings
- Read the conclusions
- Glance over the references

Able to answer the five C's

- **Category:** What type of paper is this? Measurement, theory, system, protocol, algorithm, or a survey?

- **Context:** Which other paper is it related to?

- **Correctness:** Do the assumptions appear to be valid?

- **Contributions:** What are the main contributions? Are they significant?

- **Clarity:** Is the paper well written?

Reasons NOT to read further

- Not interesting or irrelevant to my research
- Technically unsatisfied
- The assumptions appear to be invalid
- Not well written or poorly organized
- The contributions seem to be incremental

The second pass: read with greater care but not every detail (1 hour)

The second pass

- Grasp the content while ignoring technical details such as proofs and implementation

- Pay special attention to the figures, diagrams and other illustrations — they contain important information based on which the conclusions are drawn

- Mark relevant unread references for further reading

Able to summarize the main thrust

- Is the paper solving a “right” problem?
- Are the claimed contributions significant/valid with convincing supporting evidence?
- Is the approach/evaluation technically sound and novel?
- What is the potential impact of the paper?

Do I need to go to the third pass to digest the technical details?

Yes, only if

- You are interested in the technical details and have time
- You want to do some followup work
- The results are groundbreaking but somehow out of surprise or counter-intuitive
- The proof techniques, implementation details, and/or experiments turn out to be useful

The third pass: *virtually re-implement* the paper (several hours or days)

Recap

- **The first pass** (5 - 10 min): get the general idea of the paper

- If needed, go to **the second pass** (1 hour): grasp the paper's content, but not details

- If needed, go to **the third pass** (several hours): *virtually re-implement* the ideas and technical details

Advanced Cloud Computing

Cloud Concepts and Foundations

Wei Wang
CSE@HKUST
Spring 2022

THE DEPARTMENT OF
COMPUTER SCIENCE & ENGINEERING
計算機科學及工程學系

Outline

- Real-world examples of the cloud
- Definitions of cloud computing
- Key cloud concepts and characteristics
- Deployment scenarios and cloud benefits
- Introduction to AWS

Cloud: Massive Scale

- Facebook [GigaOM, 2012]
  - 30K in 2009 -> 60K in 2010 -> 180K in 2012
- Microsoft [DC knowledge]
  - > 1 million, 2013
- AWS EC2 [Randy Bias, 2009]
  - 40K, 8 cores per machine
- Google [DC knowledge]
  - > 900 K, 2013

Datacenter: outside

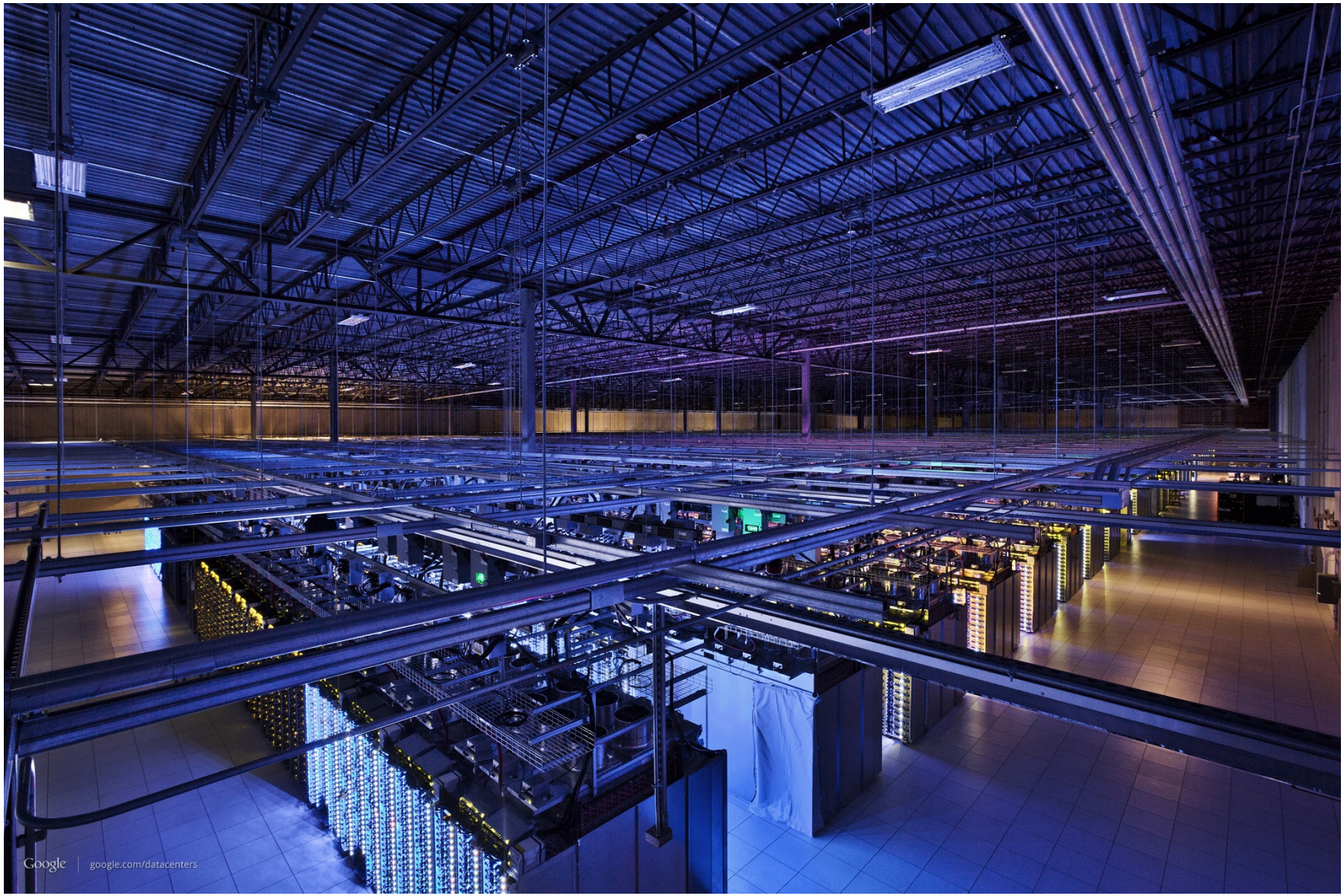
Copyright: Google

Datacenter: outside

Copyright: Google

A bird's-eye view of DC

Datacenter: inside



Copyright: Google

Server racks

Photo credit: Google

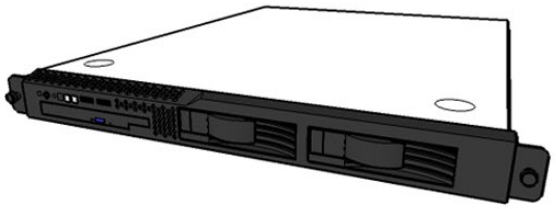
When the nights come…

Copyright: Google

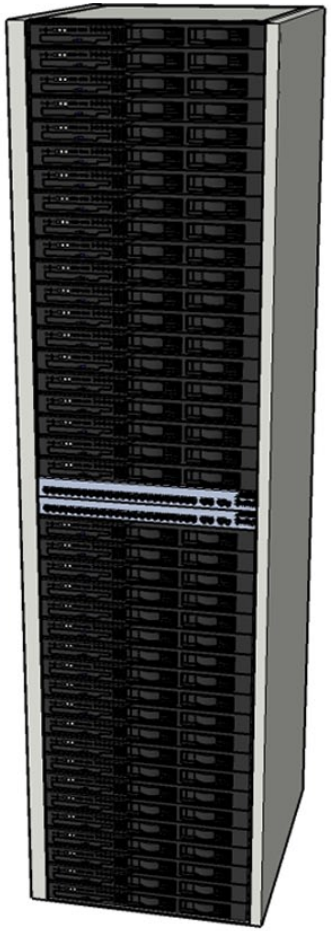
Server: inside

Server cage

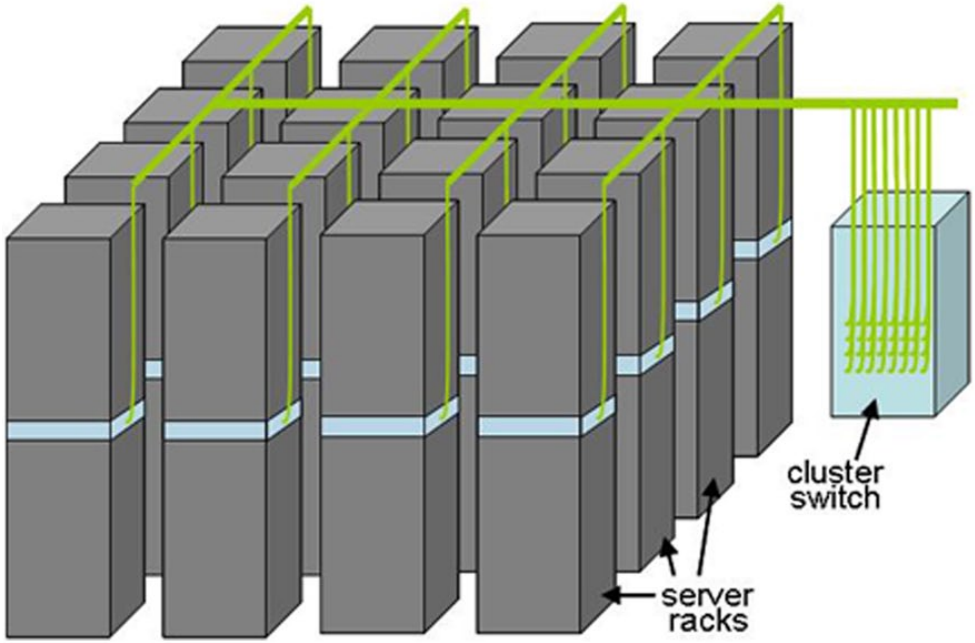
A look into the datacenter



Commodity Server



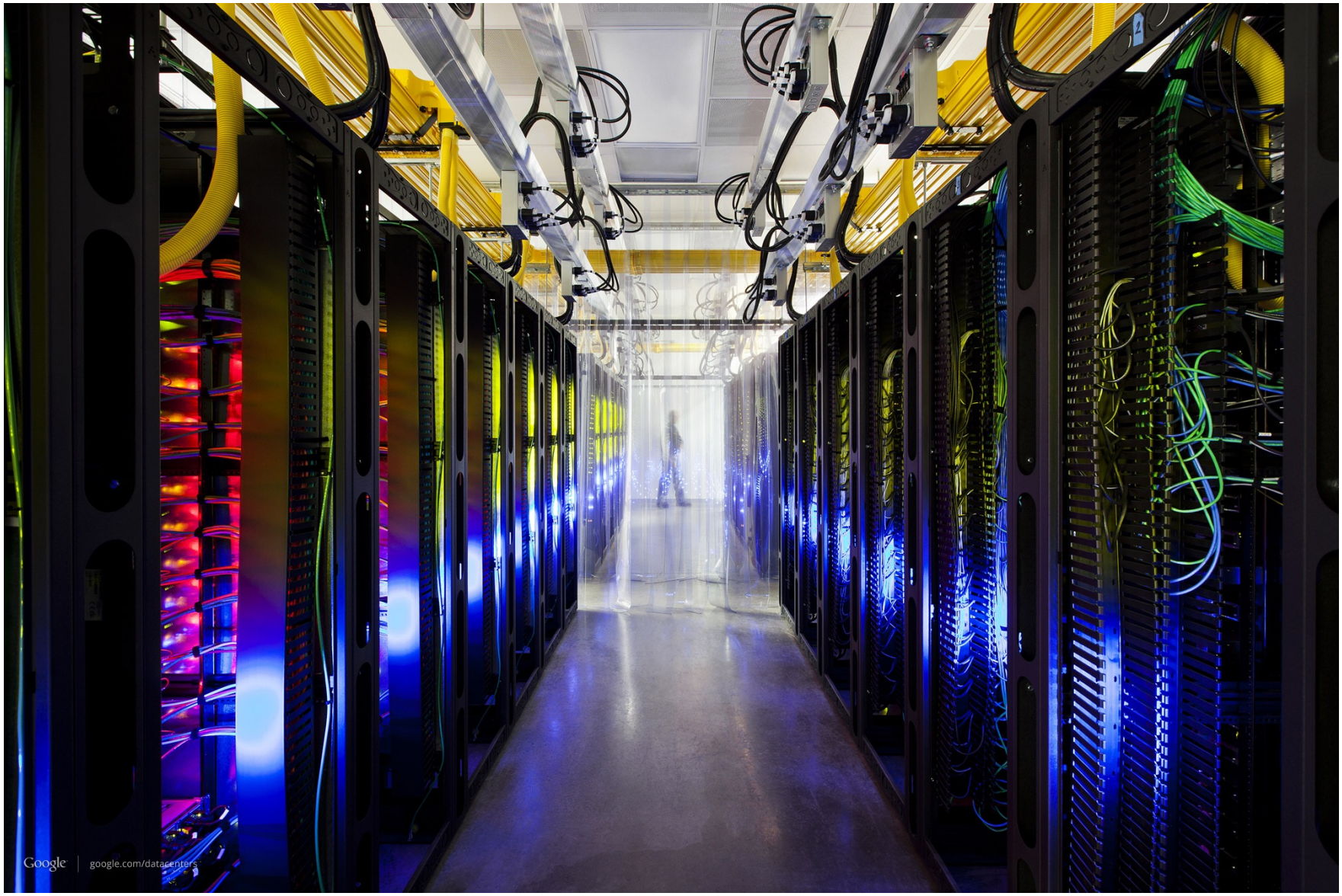
Rack



Cell

Source: L. Barroso et al., “The datacenter as a computer: An introduction to the design of warehouse-scale machines.”

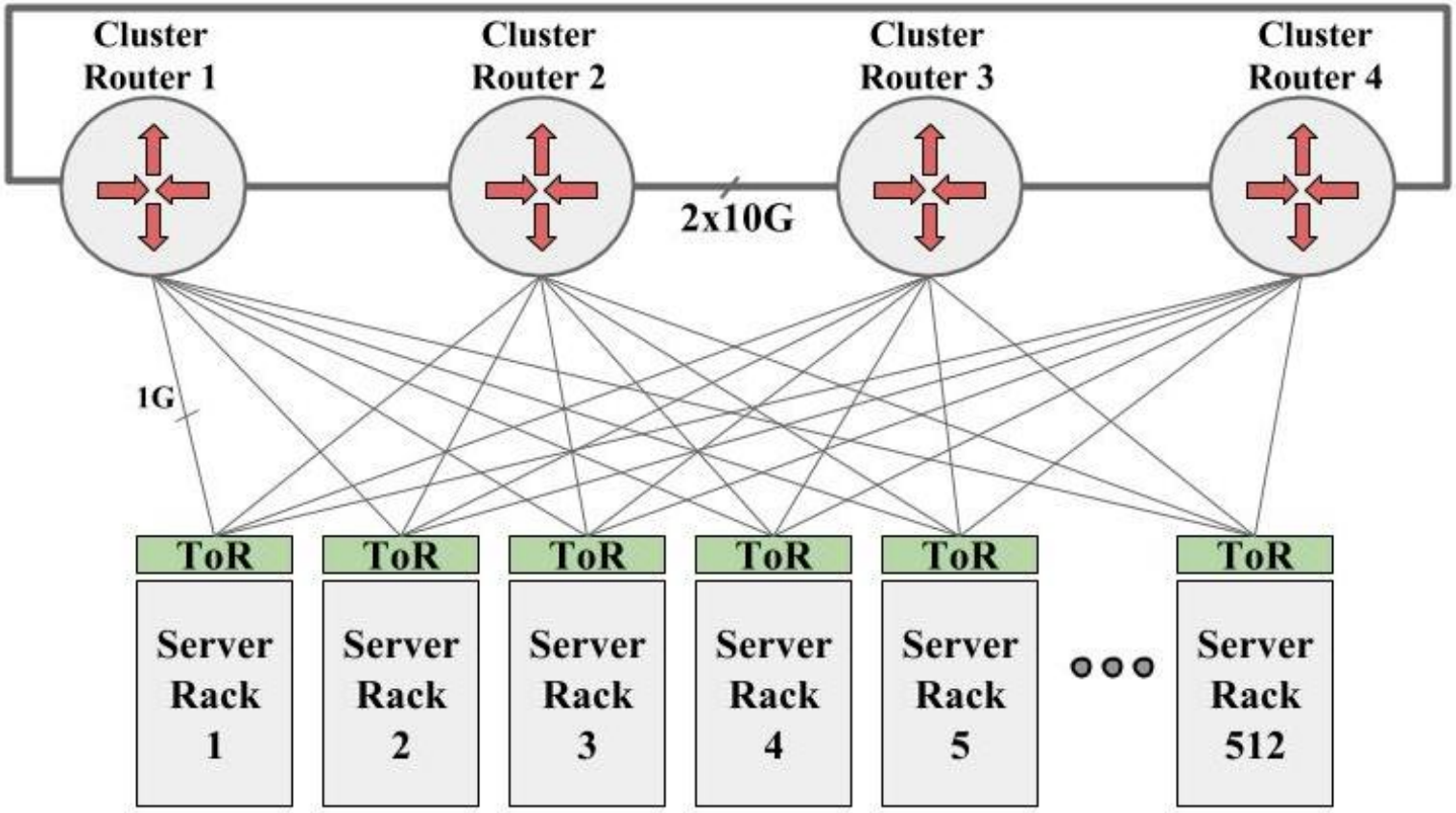
Network room



Copyright: Google

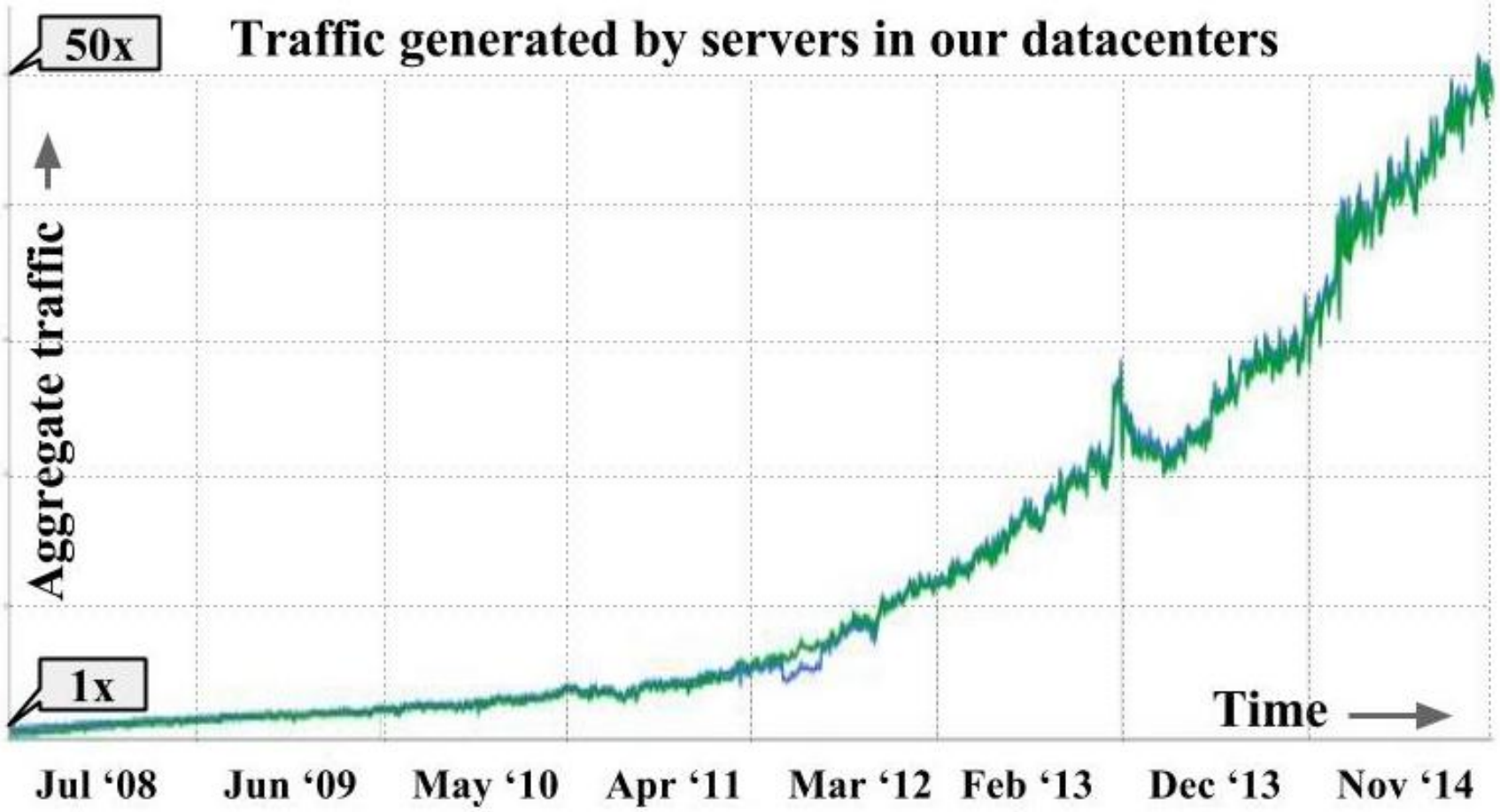
Network for a small-sized cluster

- Back to 2004 when Google has only 20k servers in a datacenter



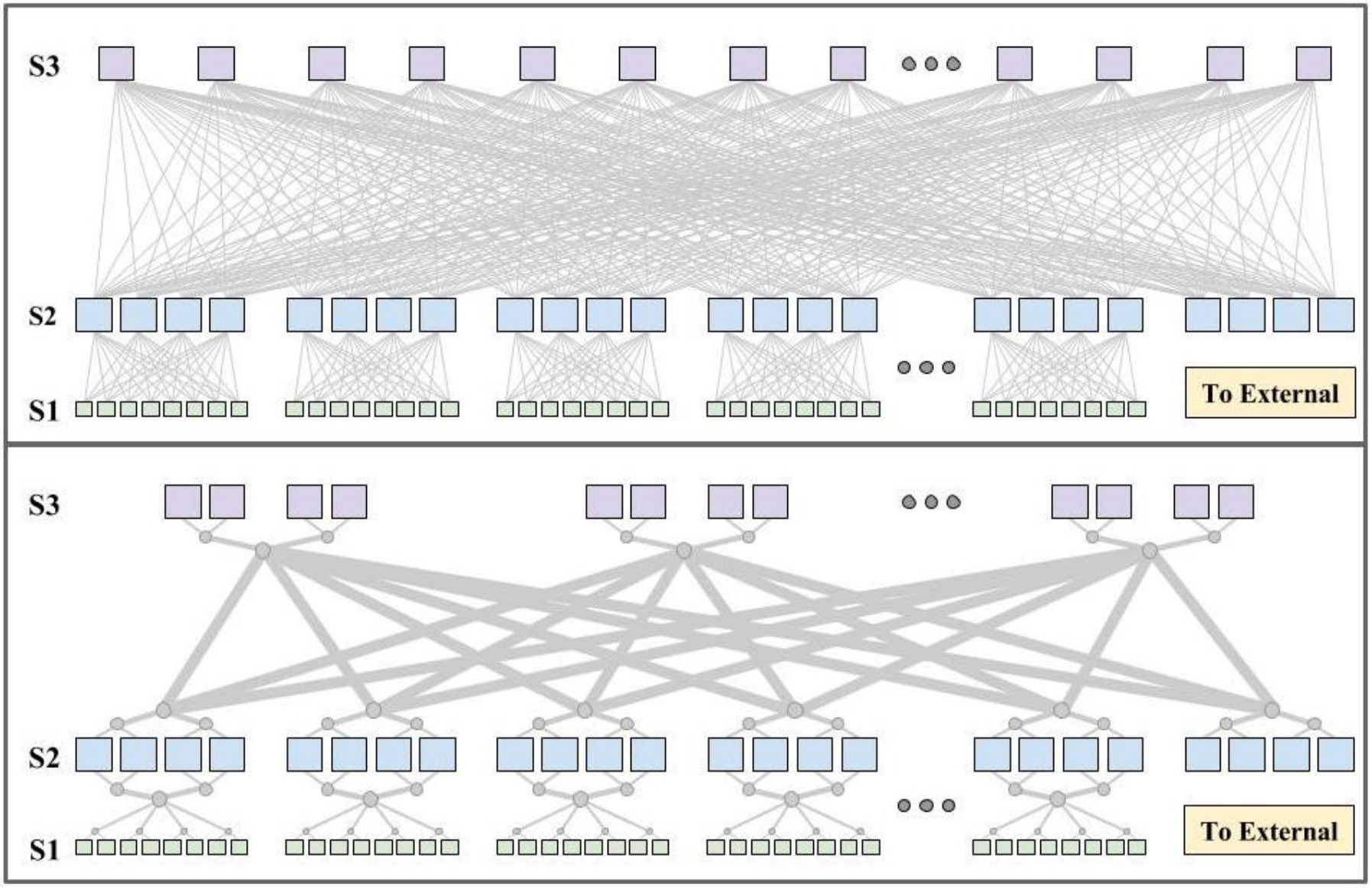
Source: A. Singh et al., "Jupiter rising: A decade of Clos topologies and centralized control in Google's datacenter network," ACM SIGCOMM'15

Things have changed quite a lot

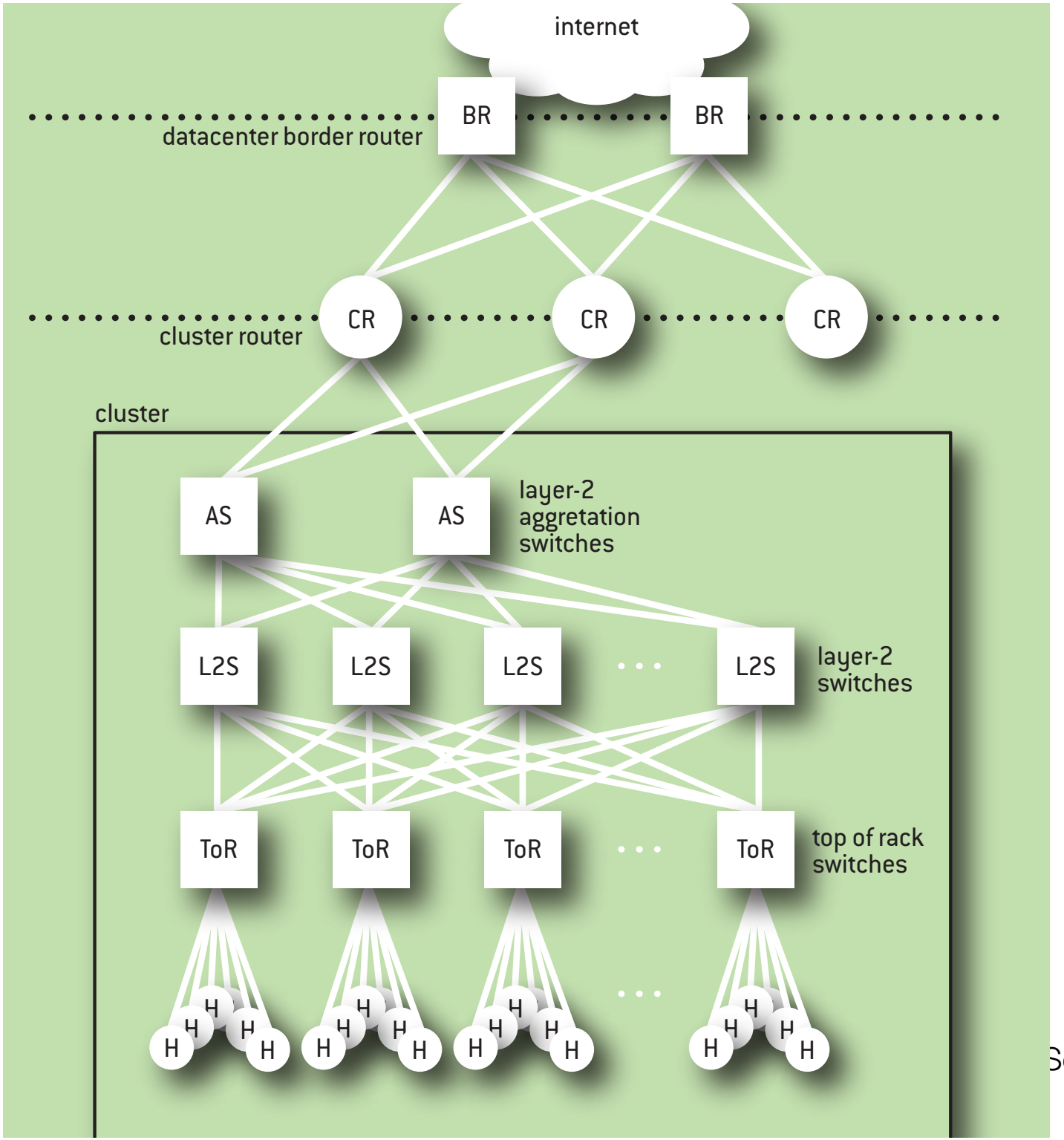


Source: A. Singh et al., "Jupiter rising: A decade of Clos topologies and centralized

When scaled up



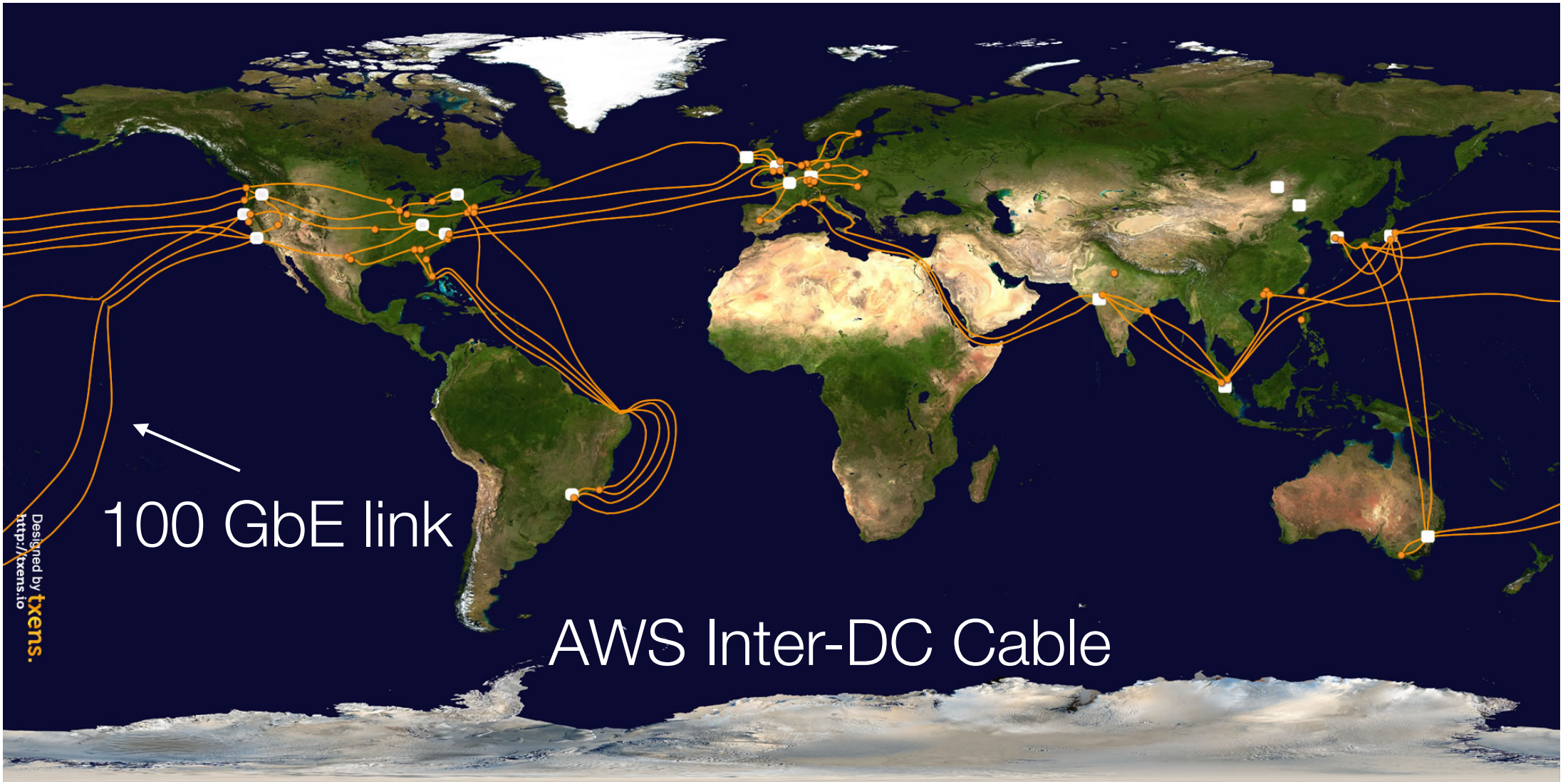
Source: A. Singh et al., “Jupiter rising: A decade of Clos topologies and centralized control in Google’s datacenter network,” ACM SIGCOMM’15



Tree-like DC Network

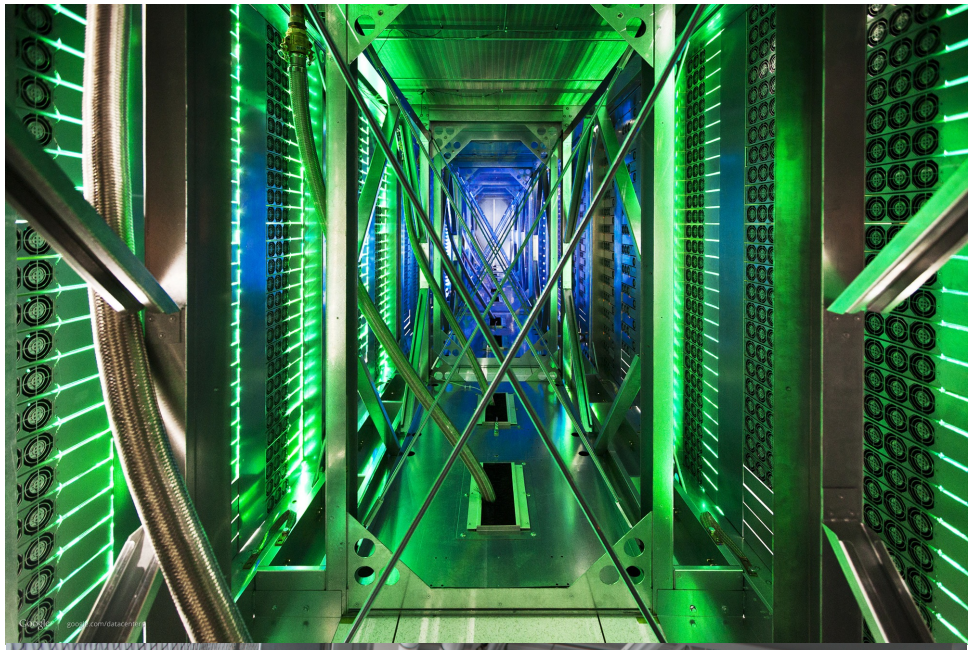
Source: http://queue.acm.org/detail.cfm?id=2208919

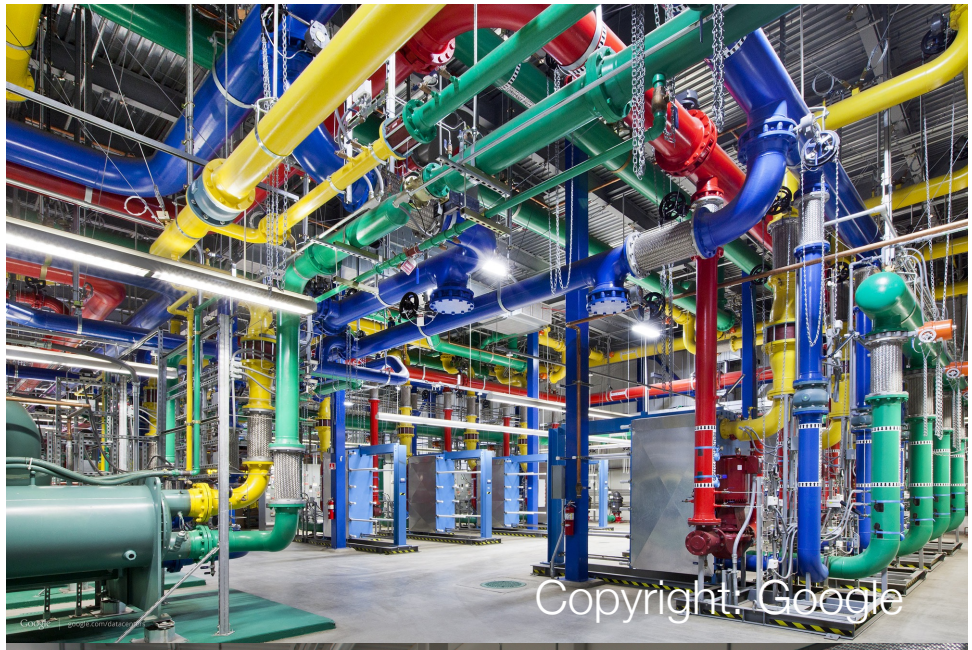
Inter-DC WAN



Source: txens, http://txens.io/index.php/2016/12/16/a-deep-dive-into-the-aws-network/

Cooling



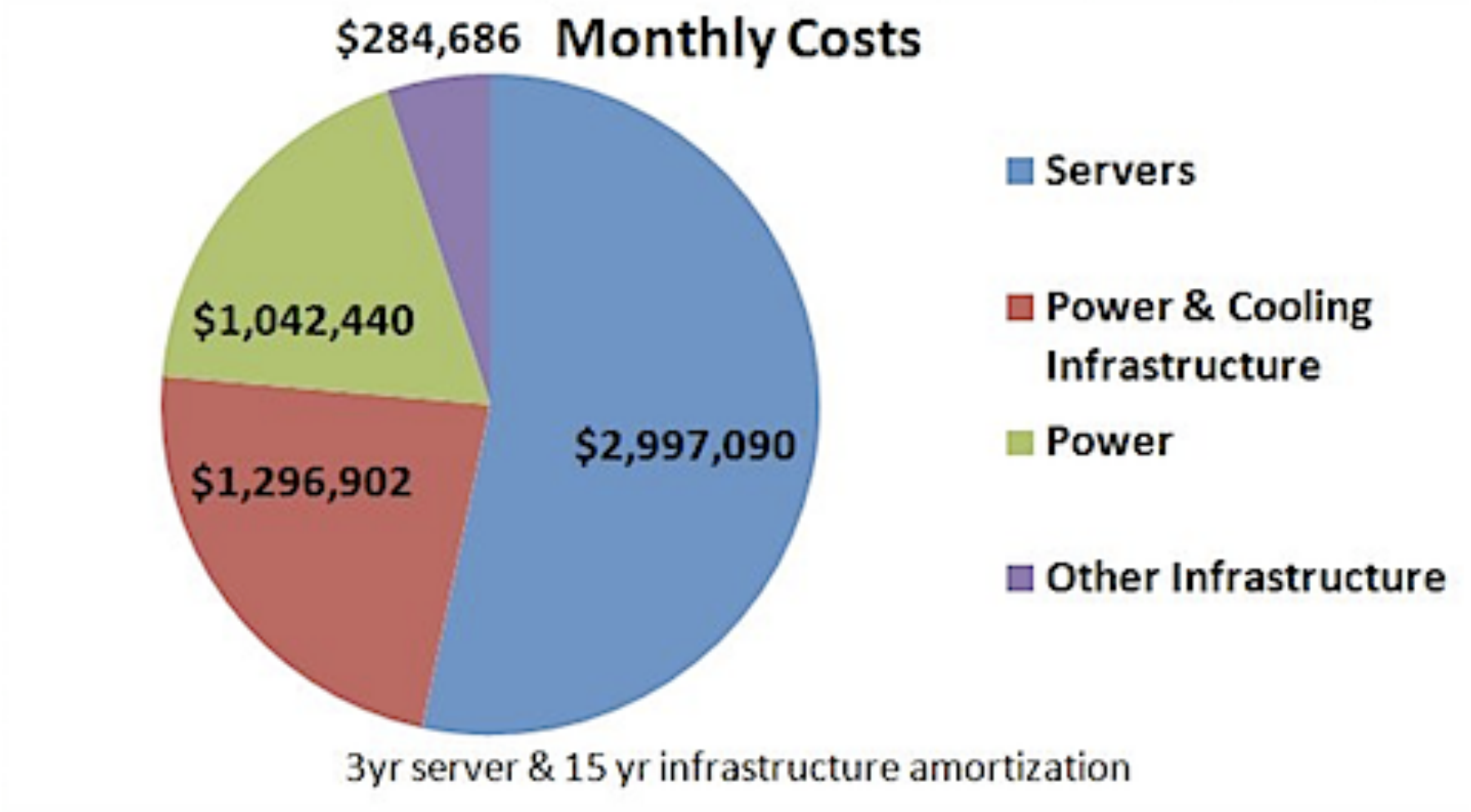


Power

Copyright: GigaOM

Copyright: Nation of Change

Cost Structure



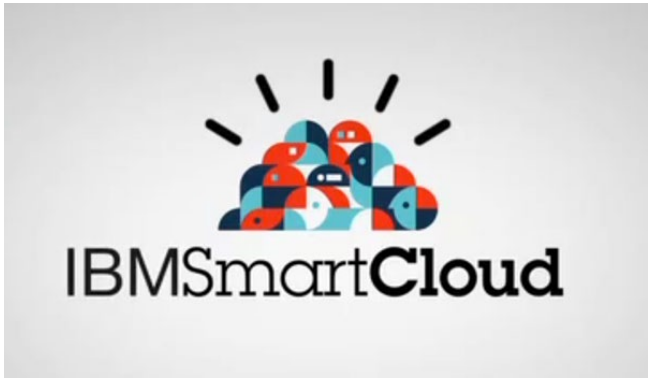
Source: James Hamilton, http://perspectives.mvdirona.com/2008/11/cost-of-power-in-large-scale-data-centers/

Explore Google Datacenter

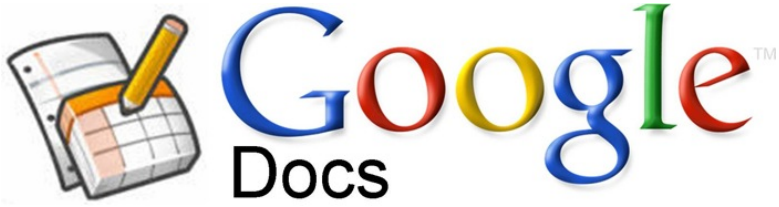
- https://www.youtube.com/watch?v=avP5d16wEp0

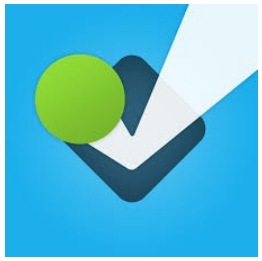
Cloud providers

- 



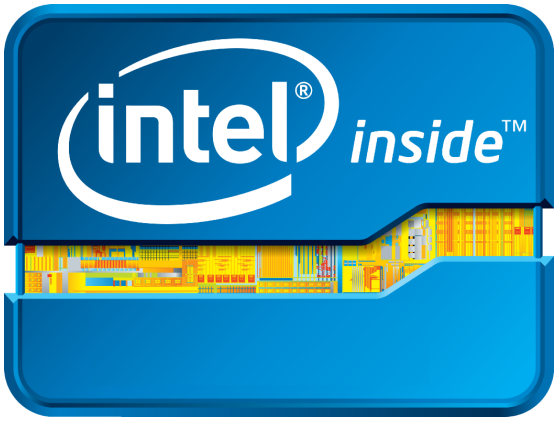
Cloud-based services

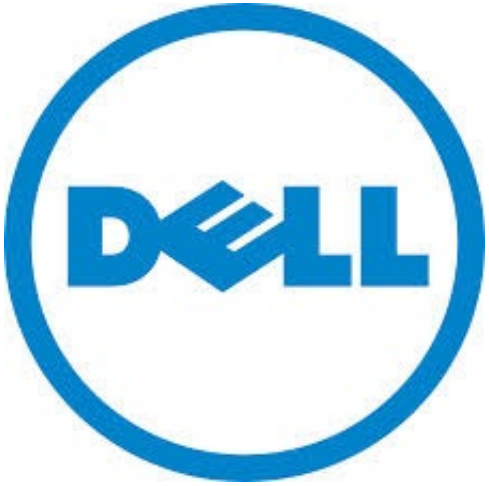




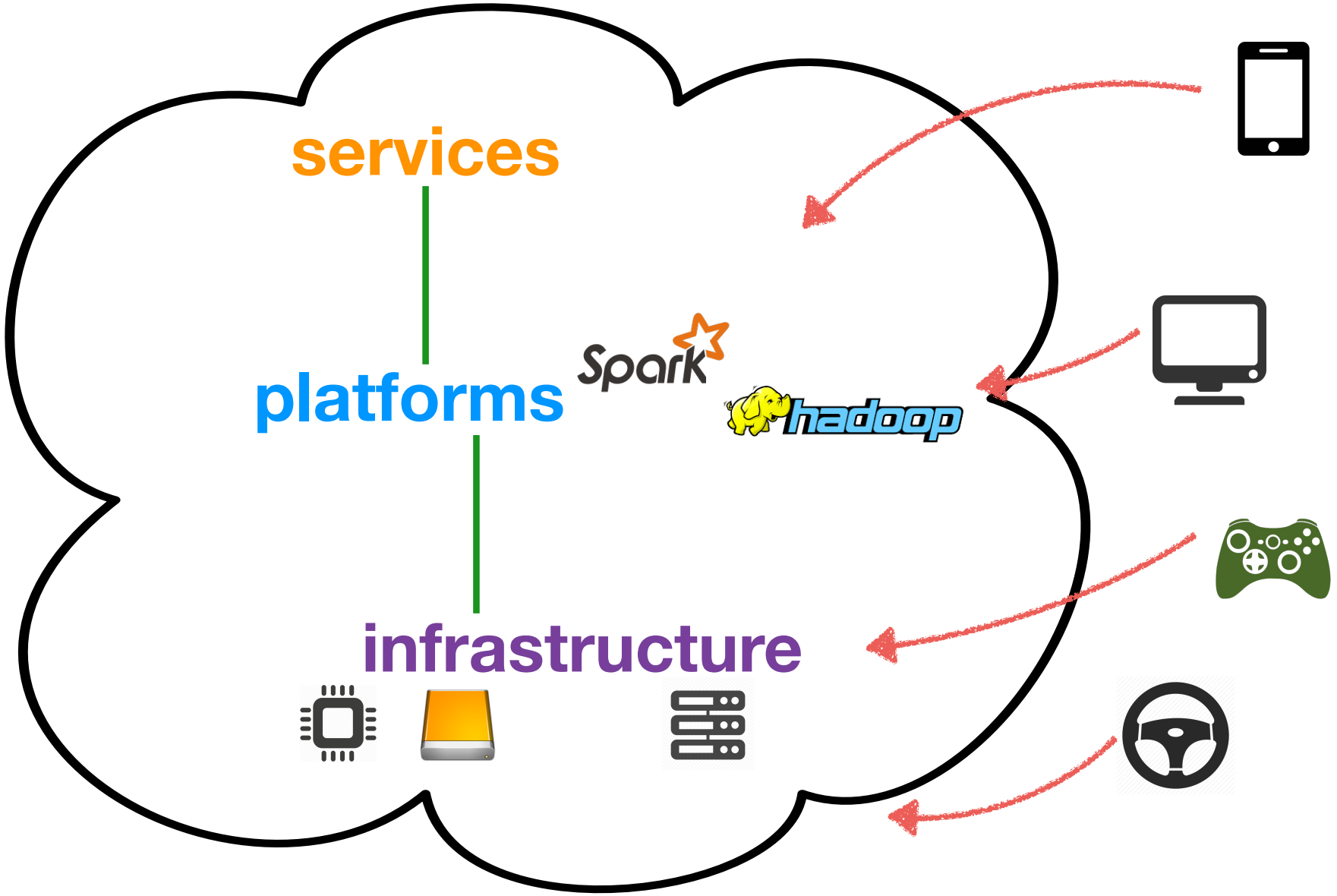
Social networking

Cloud vendors





So what is a cloud?



A long definition

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.

National Institute of Standards and Technology (NIST), U.S. Department of Commerce

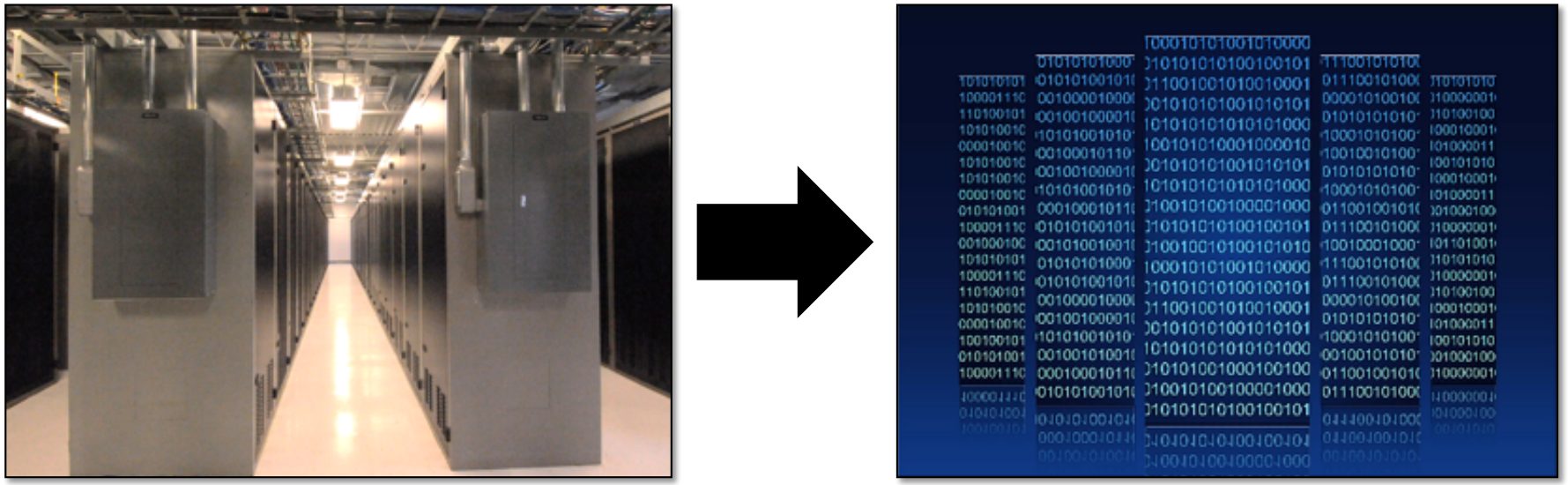
On-demand computing delivered over the **Internet** with **pay-as-you-go** pricing

Utility Computing

- Computing as the 5th utility (after water, electricity, gas, and telephony)
  - Applications and computing resources delivered as a service over the Internet
  - Pay-as-you-go
- Provided by the hardwares and system softwares hosted in the datacenters

Infrastructure as software

- Cloud computing enables users to **stop thinking of infrastructure as hardware**
- Instead, **think of (and use) it as software**



Amazon EC2

https://youtu.be/TsRBftzZsQo

1 instance runs 1000 h =
1000 instances run 1 h

Revolutionary!

Suppose you open a startup and need 100 servers

What would you do traditionally, in a pre-cloud era?

Traditional computing model

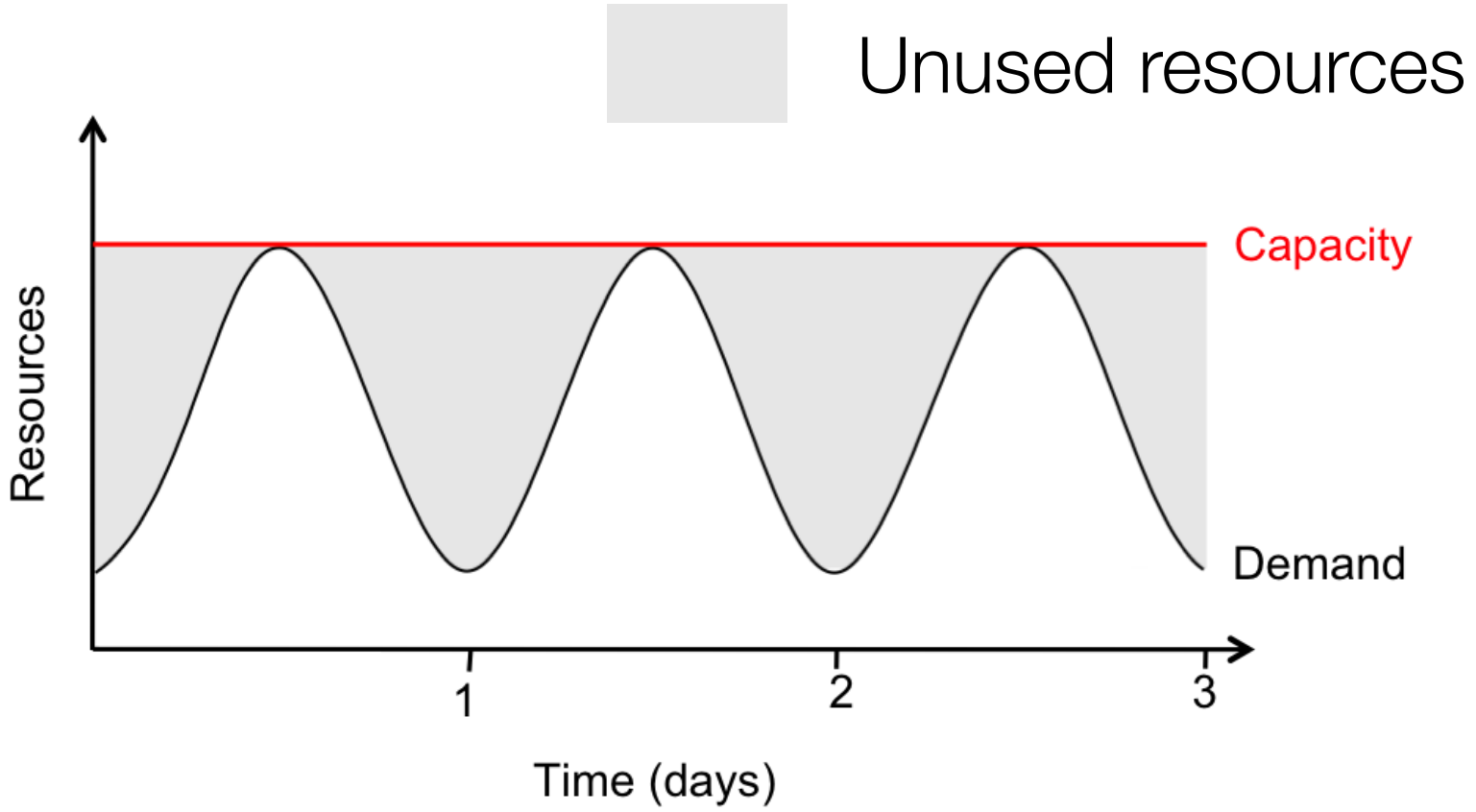
- Infrastructure as hardware
- Hardware solutions:
  - require space, staff, physical security, planning, capital expenditure
  - have a long hardware procurement cycle
  - need to provision capacity by guessing theoretical maximum peaks

Cloud computing model

- Infrastructure as software
- Flexible software solutions:
  - no upfront infrastructure investment
  - can change more quickly, easily, and cost-effectively than hardware solutions
  - eliminate the undifferentiated heavy-lifting tasks and labor works
  - Always-on services

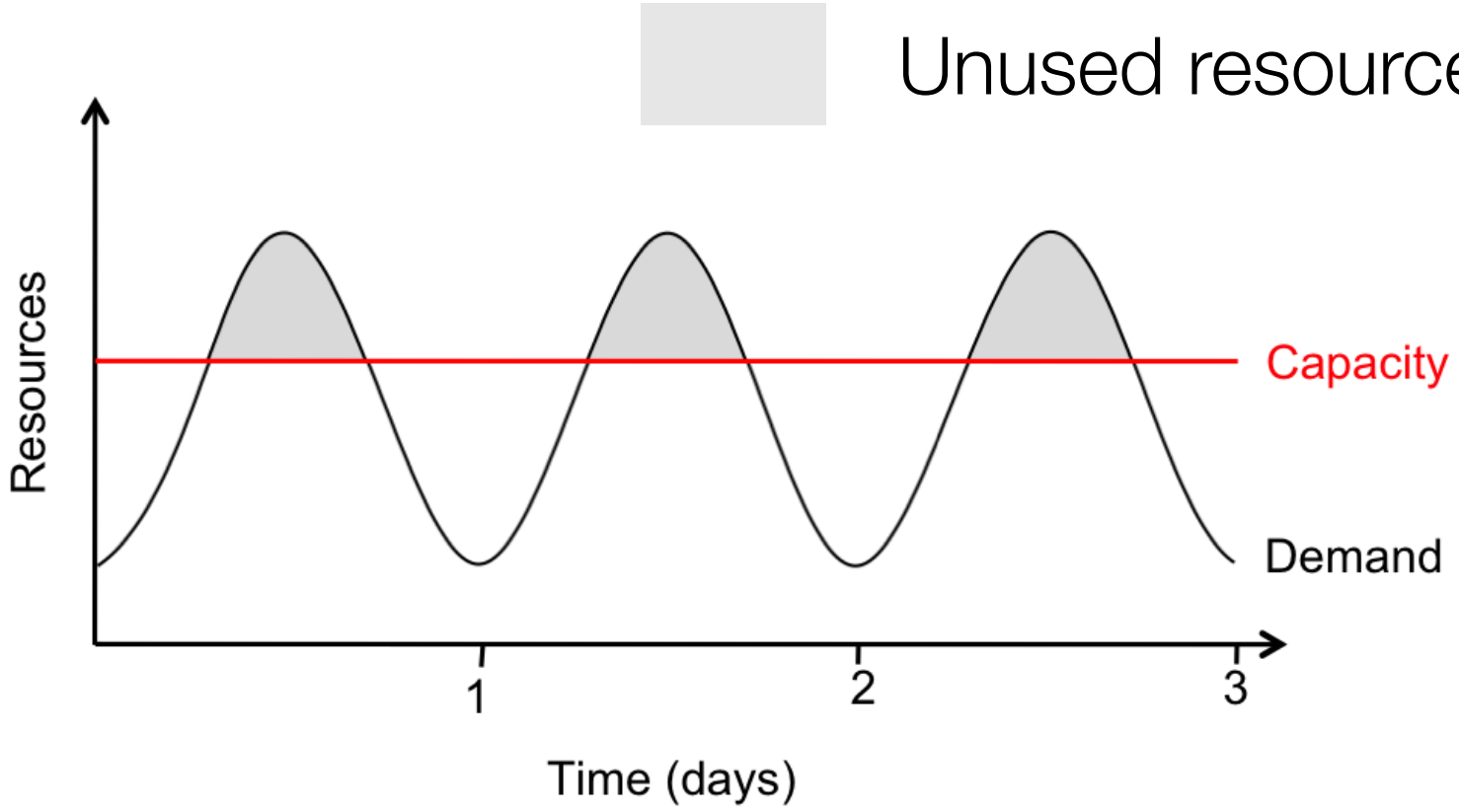
The demands are *elastic* — the 100 servers are only needed in peak time

Provisioning for peak load



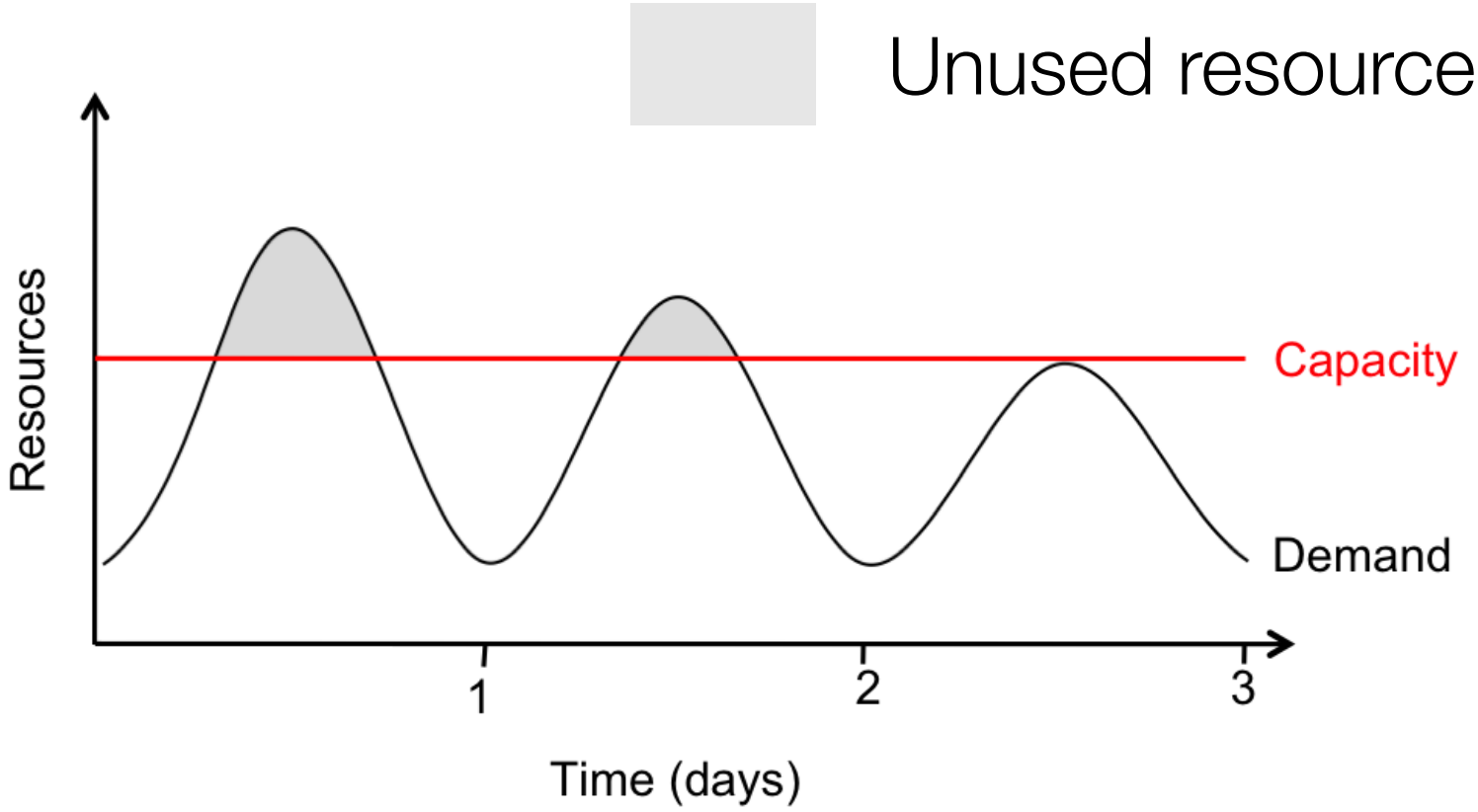
Source: Ambrust et al., “Above the clouds: A Berkeley’s view of Cloud Computing.”

Underprovisioning



Source: Ambrust et al., “Above the clouds: A Berkeley’s view of Cloud Computing.”

Underprovisioning

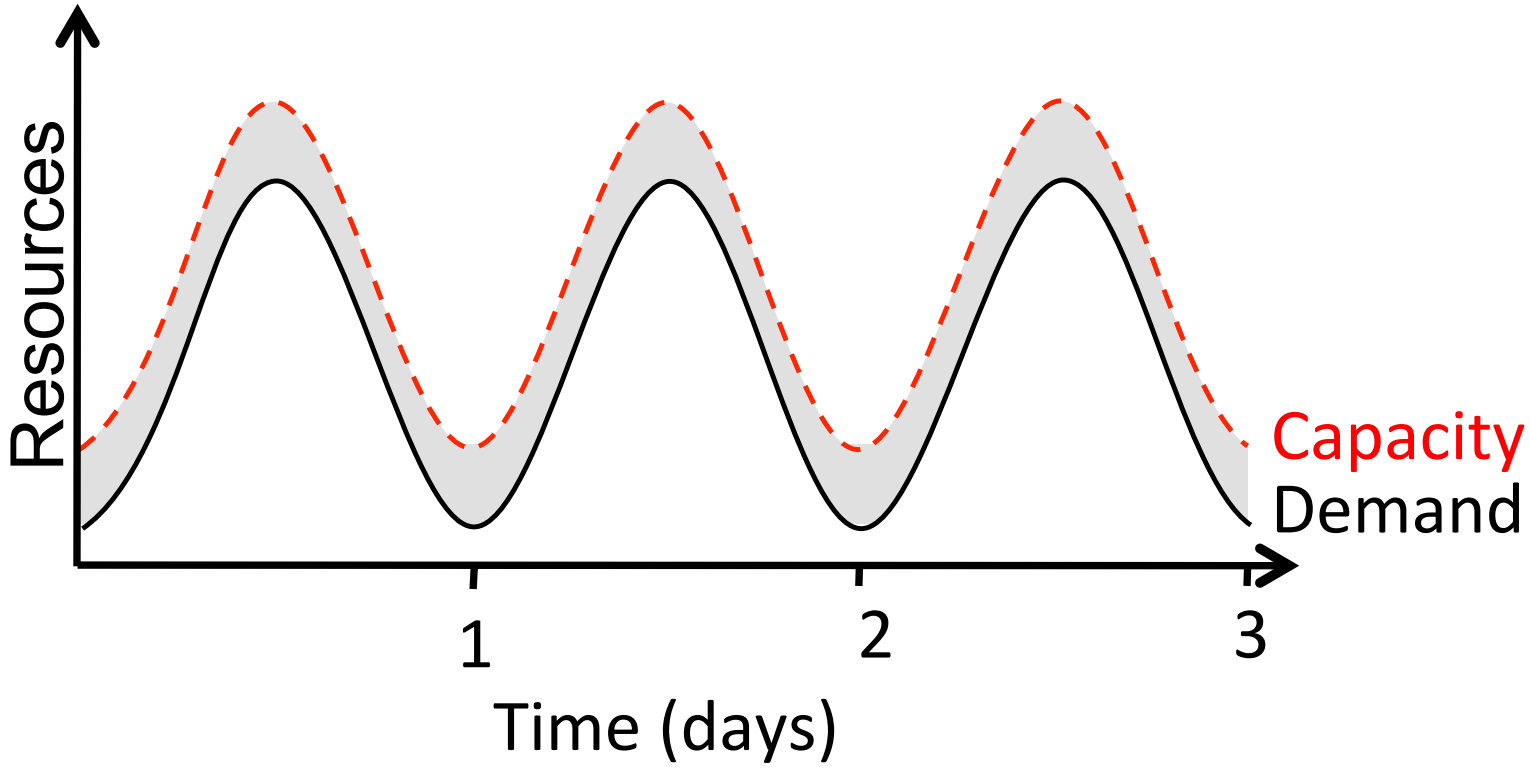


Source: Ambrust et al., “Above the clouds: A Berkeley’s view of Cloud Computing.”

Let’s do it in cloud!

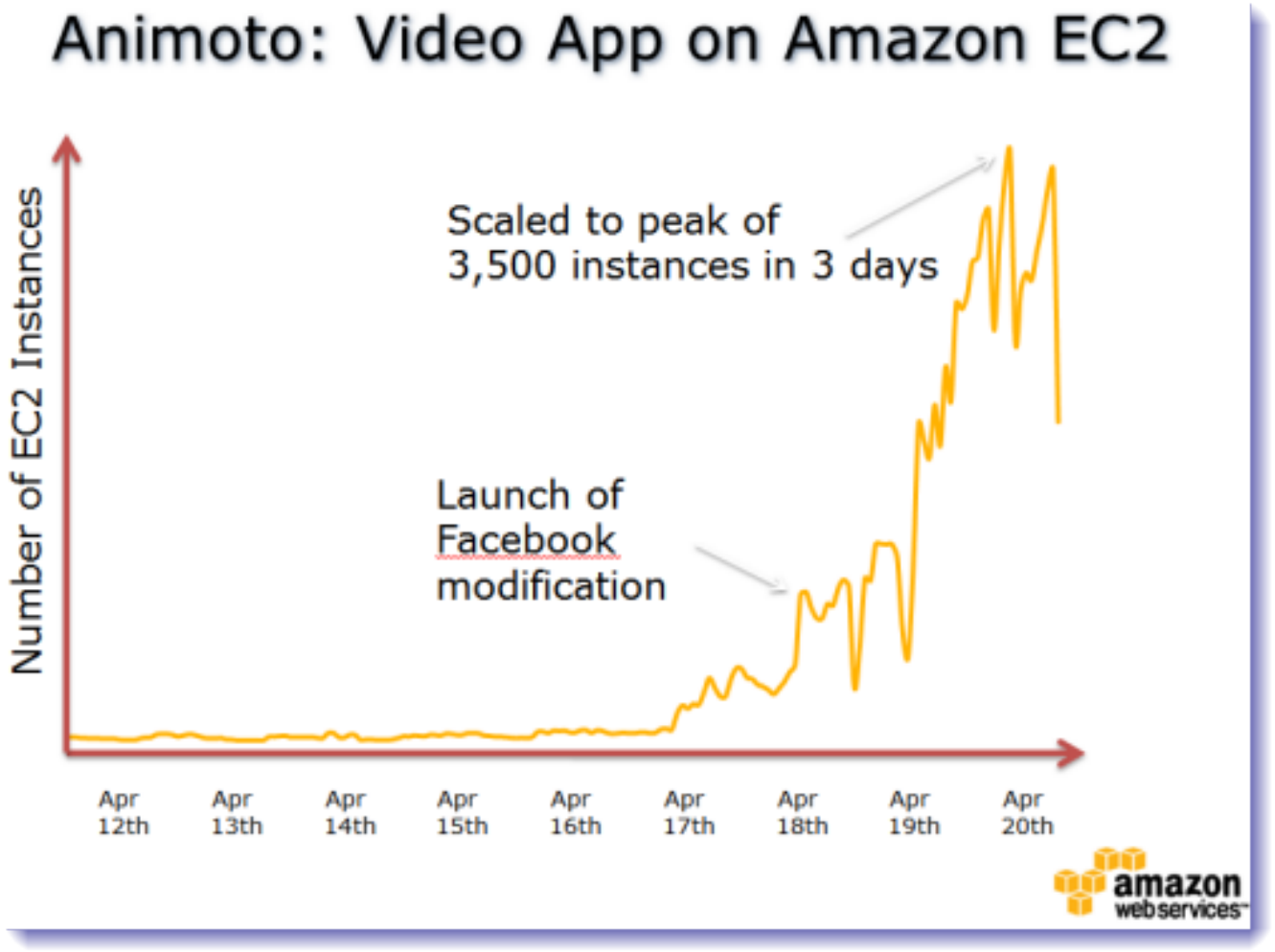
Cloud provisioning on demand

- Pay for what you used



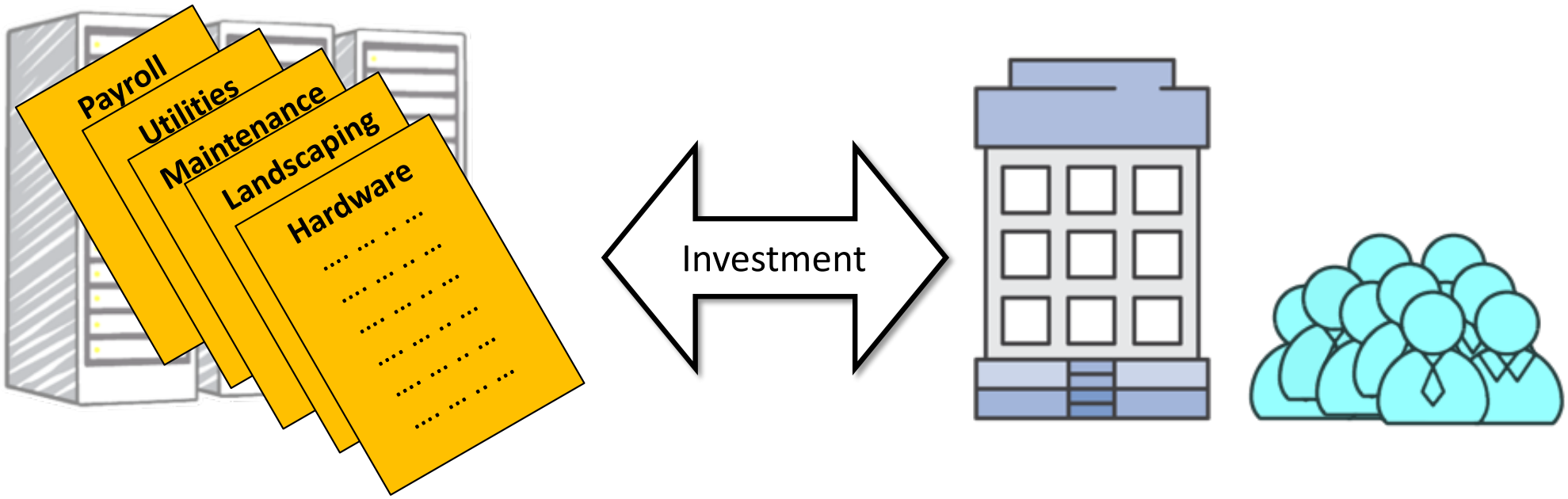
An Animoto Case Study

https://youtu.be/VwDS6MexKEo



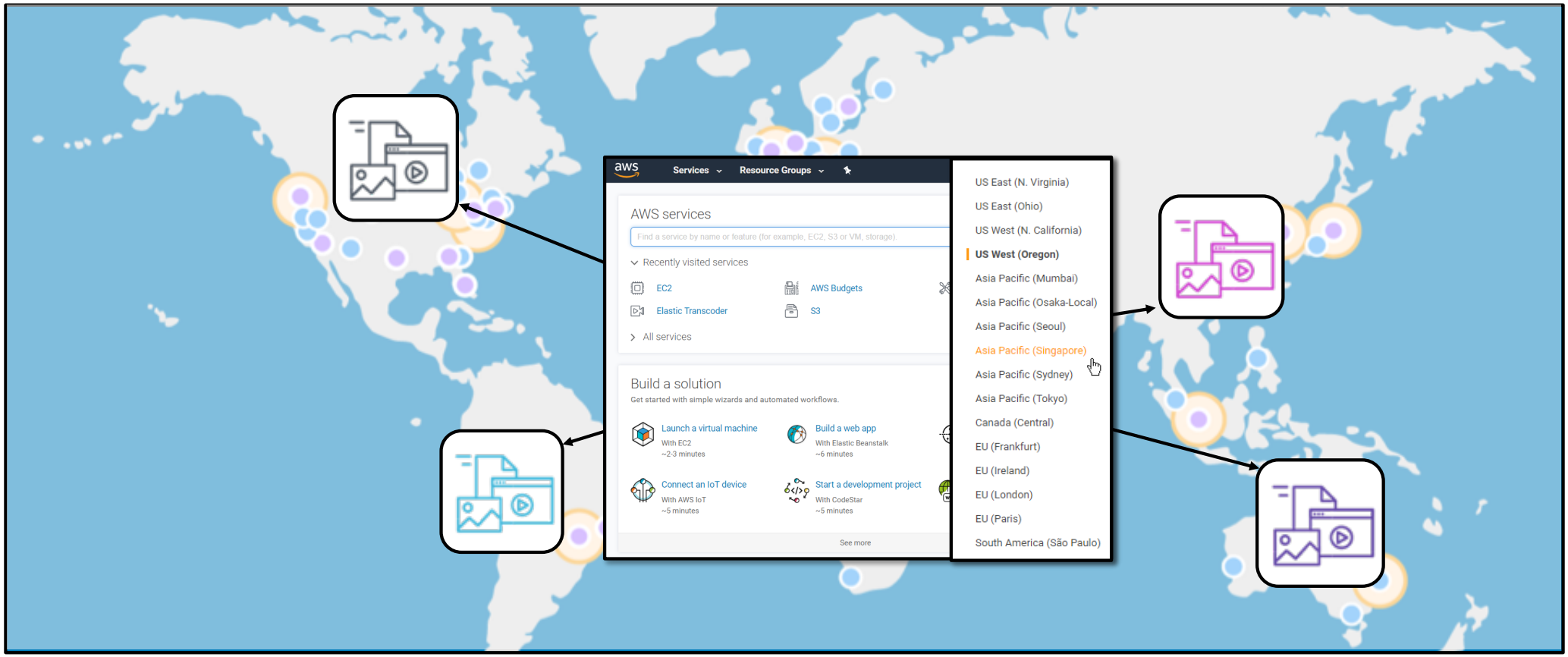
Other benefits enabled by cloud

Stop spending money on running and maintaining datacenters



Other benefits enabled by cloud

Go global in minutes



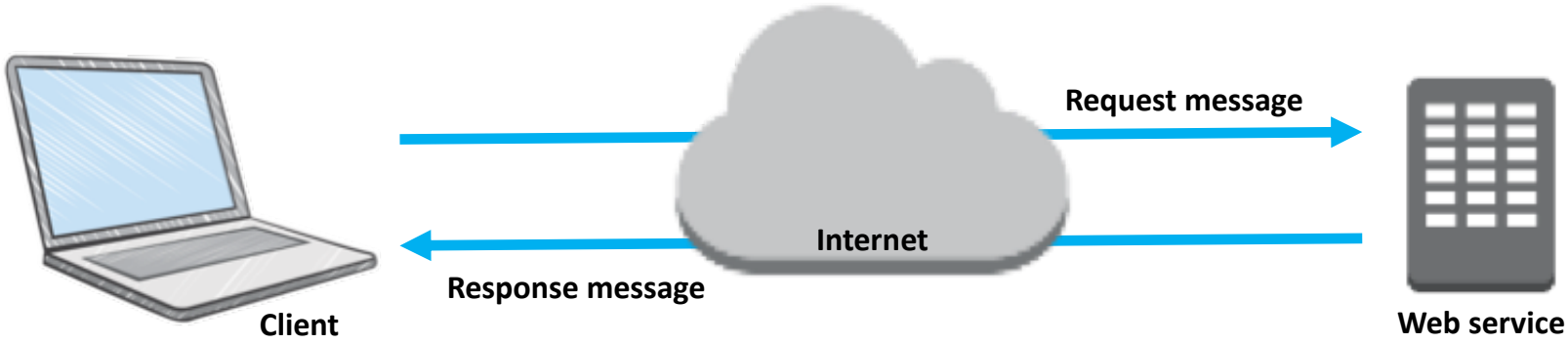
Summary: Why cloud?

- Better capital utilization
- The unit cost of on-demand capacity may be higher than the unit cost of fixed capacity; offset by no charge when capacity is not being used
- Elasticity, easy to scale up and down
- Access to complex infrastructure and resources without internal resources

Amazon Web Services

What are web services?

- A piece of software that makes itself available over the internet and uses a **standardized format**—such as XML or JSON—for the request and the response of an **application programming interface (API) interaction**.



What is AWS?

- AWS is **a secure cloud platform** that offers **a broad set of global cloud-based products**

- provides **on-demand access** to compute, storage, network, database, and other IT resources and management tools

What is AWS?

- AWS is **a secure cloud platform** that offers **a broad set of global cloud-based products**
- AWS provides **on-demand access** to compute, storage, network, database, and other IT resources and management tools
- AWS offers **flexibility** and **pay-as-you-go pricing**
- AWS services work together like building blocks

Categories of AWS services

Compute services –
- Amazon EC2
- AWS Lambda
- AWS Elastic Beanstalk
- Amazon EC2 Auto Scaling
- Amazon ECS
- Amazon EKS
- Amazon ECR
- AWS Fargate

Security, Identity, and Compliance services –
- AWS IAM
- Amazon Cognito
- AWS Shield
- AWS Artifact
- AWS KMS

Storage services –
- Amazon S3
- Amazon S3 Glacier
- Amazon EFS
- Amazon EBS

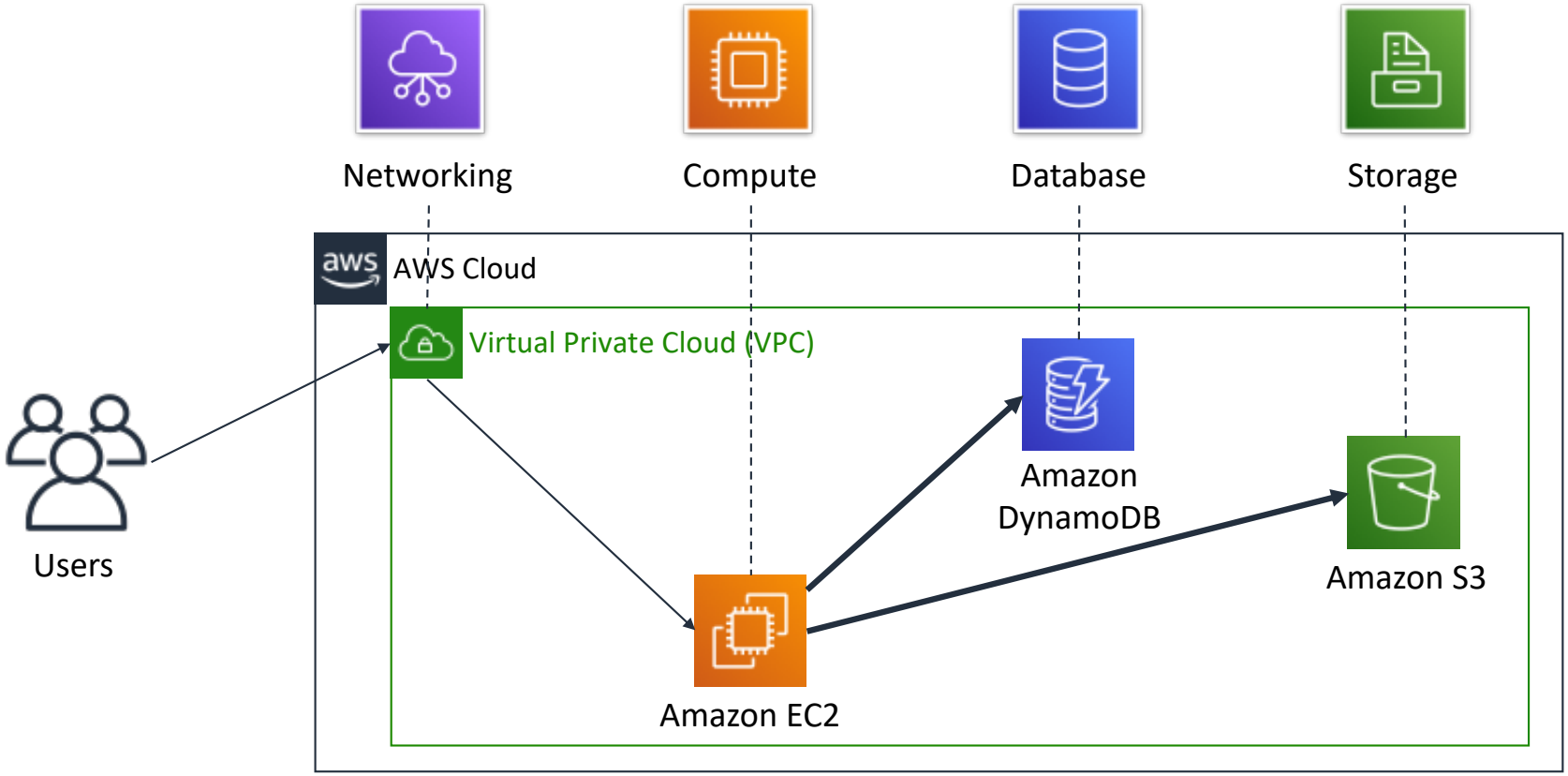
Database services –
- Amazon RDS
- Amazon DynamoDB
- Amazon Redshift
- Amazon Aurora

Networking and Content Delivery services –
- Amazon VPC
- Amazon Route 53
- Amazon CloudFront
- Elastic Load Balancing

Management and Governance services –
- AWS Trusted Advisor
- AWS CloudWatch
- AWS CloudTrail
- AWS Well-Architected Tool
- AWS Auto Scaling
- AWS Command Line Interface
- AWS Config
- AWS Management Console
- AWS Organizations

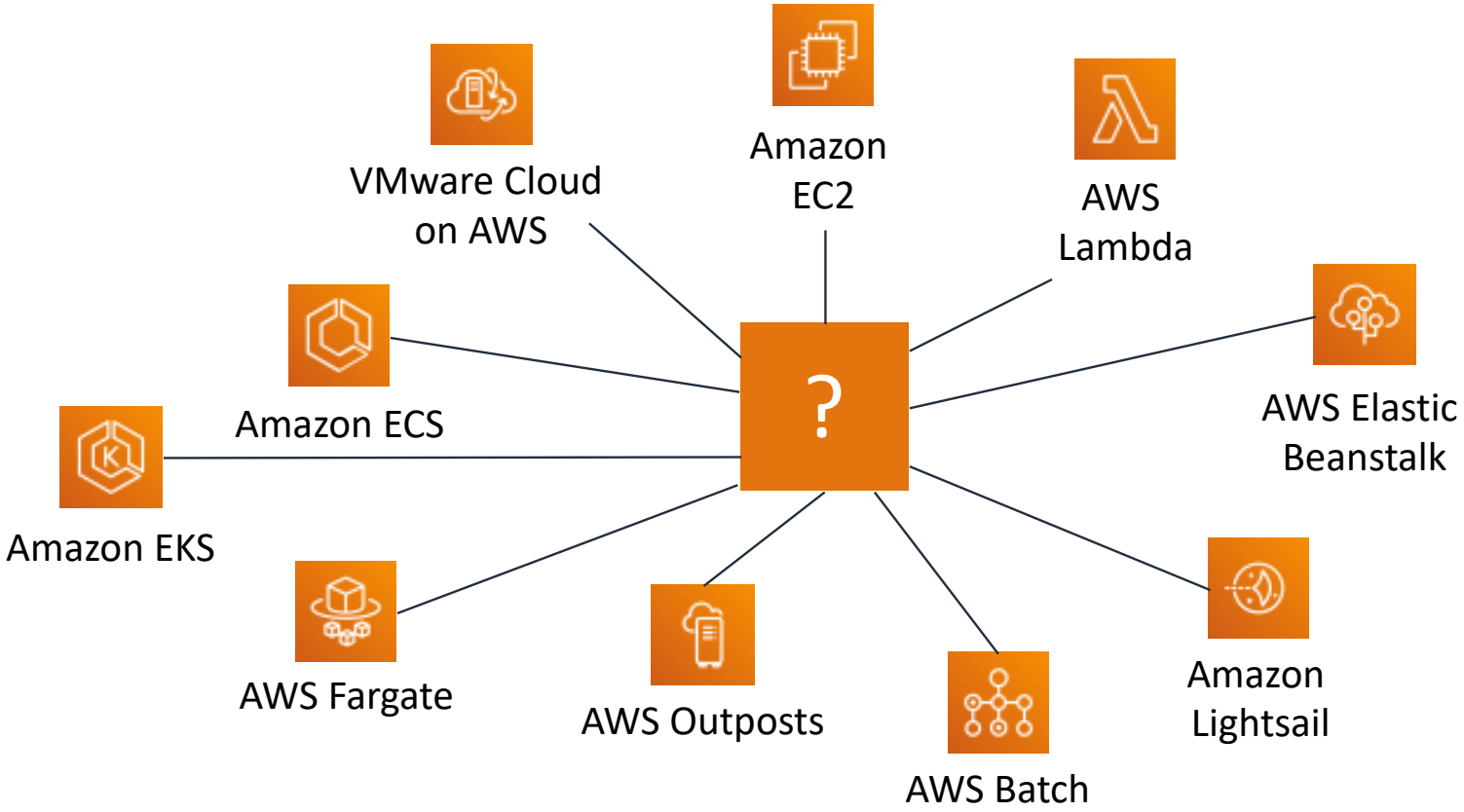
AWS Cost Management services –
- AWS Cost & Usage Report
- AWS Budgets
- AWS Cost Explorer

Simple solution example



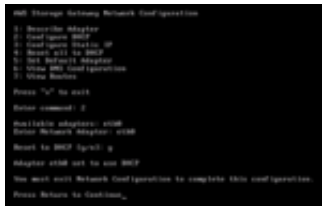
Choosing a service

- The service you choose depends on your business goals and technology requirements



3 ways to interact w/ AWS

- AWS Management Console
  - easy-to-use graphical interface



- Command Line Interface (AWS CLI)
  - access to services by discrete commands or scripts

- Software Development Kits (SDKs)
  - access to services directly from your code (e.g., Java, Python, etc.)

Credit

- Some slides are adapted from Prof. Hong Xu's slides for CS 4296/5296 in CityU
- Some slides are adapted from AWS Academy Class (Cloud foundations)