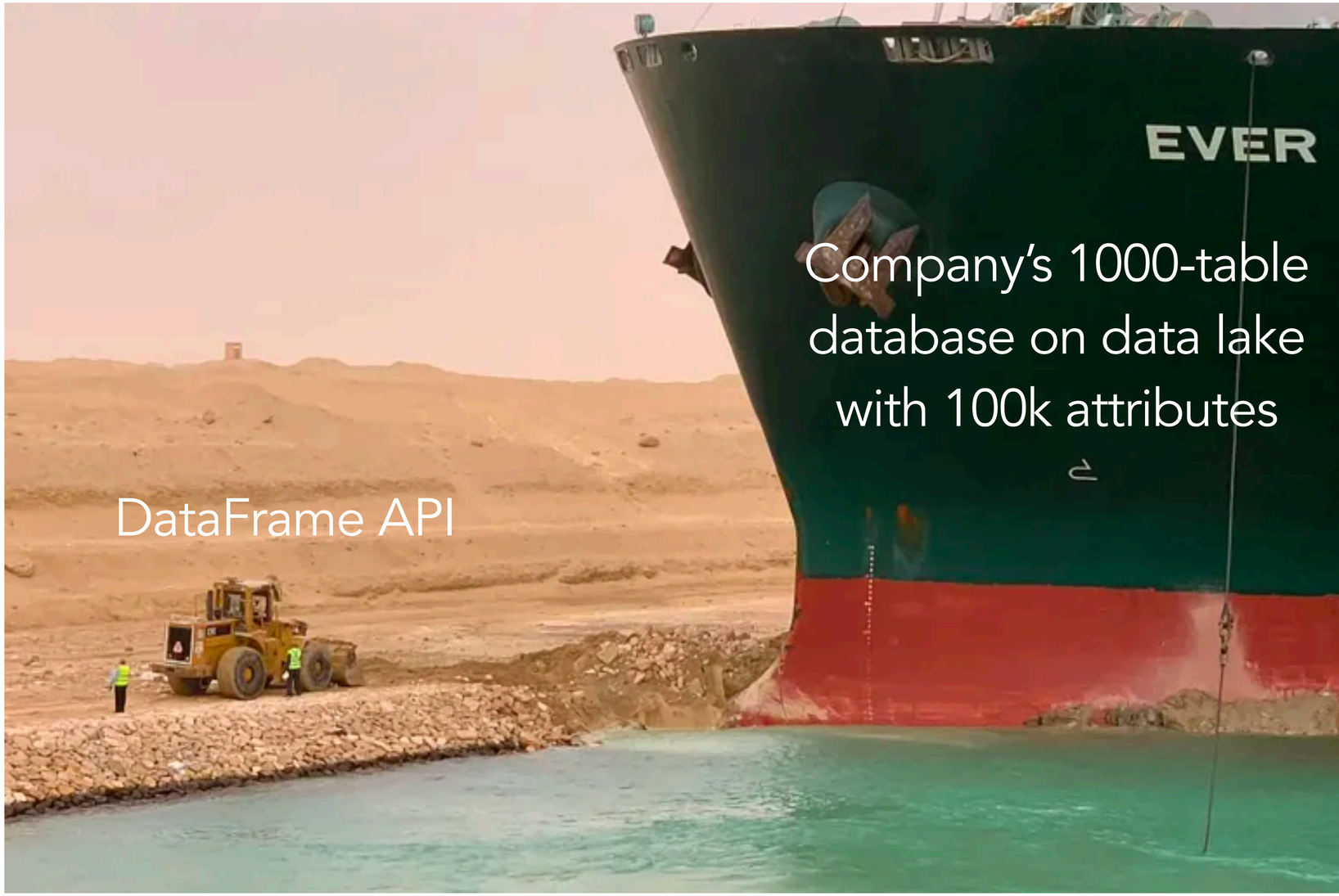
DSC 204a Scalable Data Systems

- Haojian Jin



Meme idea credit: https://datasystemsfun.tumblr.com/

Logistics

- Start your assignment early.
  - Get help from your TAs.
  - Double TA resources in this class.
- Do not write a last minute email.
  - Today is ddl. I have XXX issues. Please resolve this issue ASAP!
  - Expect TA’s response in 24 hours, excluding weekends.

Where are we in the class?

Foundations of Data Systems (2 weeks)

- Digital representation of Data → Computer Organization → Memory hierarchy → Process → Storage

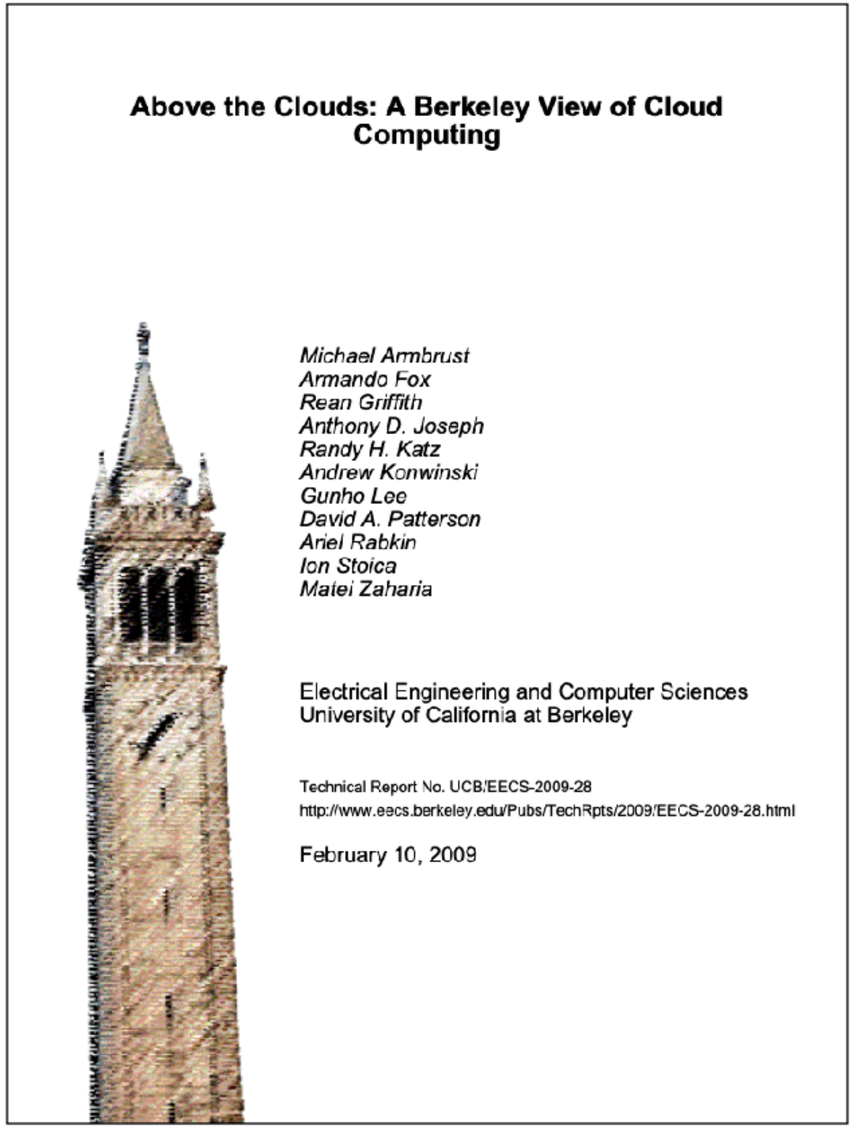
Scaling Distributed Systems (3 weeks)

- **Cloud** → Distributed storage → Partition and replication (HDFS) → Distributed computation

Data Processing and Programming model (5 weeks)

- Data Models evolution → Data encoding evolution → → IO & Unix Pipes → Batch processing (MapReduce) → Stream processing (Spark)

Suggested reading



Above the Clouds: A Berkeley View of Cloud Computing

Michael Armbrust
Armando Fox
Rean Griffith
Anthony D. Joseph
Randy H. Katz
Andrew Konwinski
Gunho Lee
David A. Patterson
Ariel Rabkin
Ion Stoica
Matei Zaharia

Electrical Engineering and Computer Sciences
University of California at Berkeley

Technical Report No. UCB/EECS-2009-28
http://www.eecs.berkeley.edu/Pubs/TechRpts/2009/EECS-2009-28.html

February 10, 2009

University of California, Berkeley
https://www2.eecs.berkeley.edu › EECS-2009-28

Above the Clouds: A Berkeley View of Cloud Computing

by M Armbrust · 2009 · Cited by 9289 — Our goal in this paper to clarify terms, provide simple formulas to quantify comparisons between of **cloud** and conventional **Computing**, and ...
25 pages

Today’s topic

- Cloud computing evolution - sharing granularity
- Cloud computing layers
- Advantages of Cloud computing

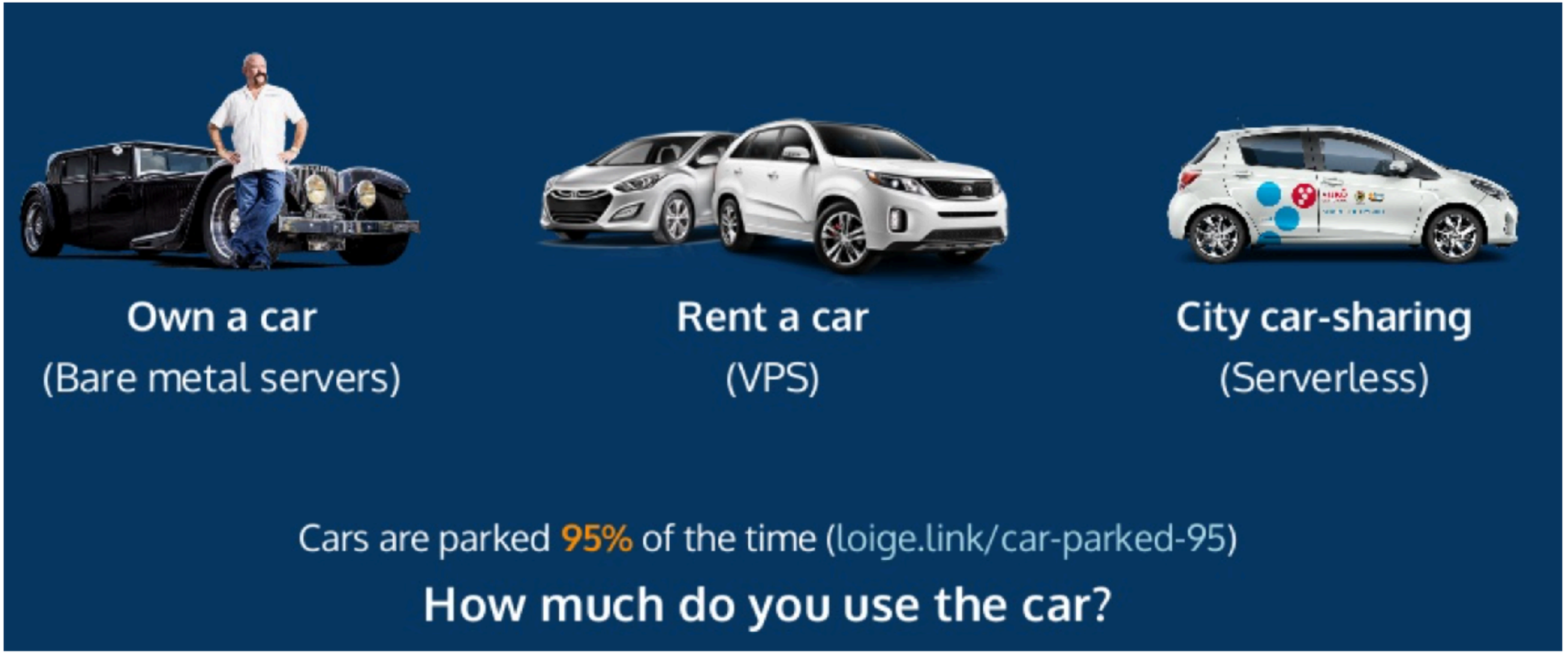
Cloud Computing

- Compute, storage, memory, networking, etc. are **virtualized** and exist on **remote** *servers*; **rented** by application users
- The opposite:
  - On-premises refers to IT infrastructure hardware and software applications that are hosted on-site.

Evolution of Cloud Infrastructure

- **Data Center**: Physical space from which a cloud is operated
- **3 generations of data centers/clouds:**
  - Cloud 1.0 (Past)
  - Cloud 2.0 (Current)
  - Cloud 3.0 (Ongoing Research)

Car Analogy



https://www.slideshare.net/loige/building-a-serverless-company-with-nodejs-react-and-the-serverless-framework-jsday-2017-verona

Cloud 1.0 (Past)

- Networked servers;
- User rents servers (time-sliced access) needed for data/software
- **Cloud 2.0 (Current)**: "Virtualization" of networked servers; user rents amount of resource capacity; cloud provider has a lot more flexibility on provisioning (multi-tenancy, load balancing, more elasticity, etc.)
- **Cloud 3.0 (Ongoing Research)**: "Serverless" and disaggregated resources all connected to fast networks

From Lecture 5:

Virtualization of Hardware Resources

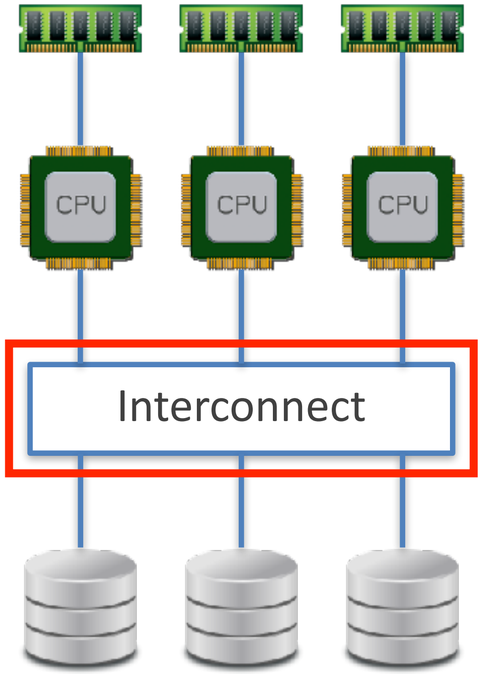
> ***Q:*** *But is it not risky/foolish for OS to hand off control of hardware to a process (random user-written program)?!*

- OS has *mechanisms* and *policies* to regain control
- **Virtualization:**
  - Each hardware resource is treated as a virtual entity that OS can divvy up among processes in a controlled way
- **Limited Direct Execution:**
  - OS mechanism to time-share CPU and preempt a process to run a different one, aka "context switch"
  - A **Scheduling policy** tells OS what time-sharing to use
  - Processes also must transfer control to OS for "privileged" operations (e.g., I/O); **System Calls** API

Cloud 2.0 (Current)

- “Virtualization” of networked servers;
- User rents amount of resource capacity (e.g., memory, disk);
- Cloud provider has a lot more flexibility on provisioning (multi-tenancy, load balancing, more elasticity, etc.)

Parallelism in the Cloud



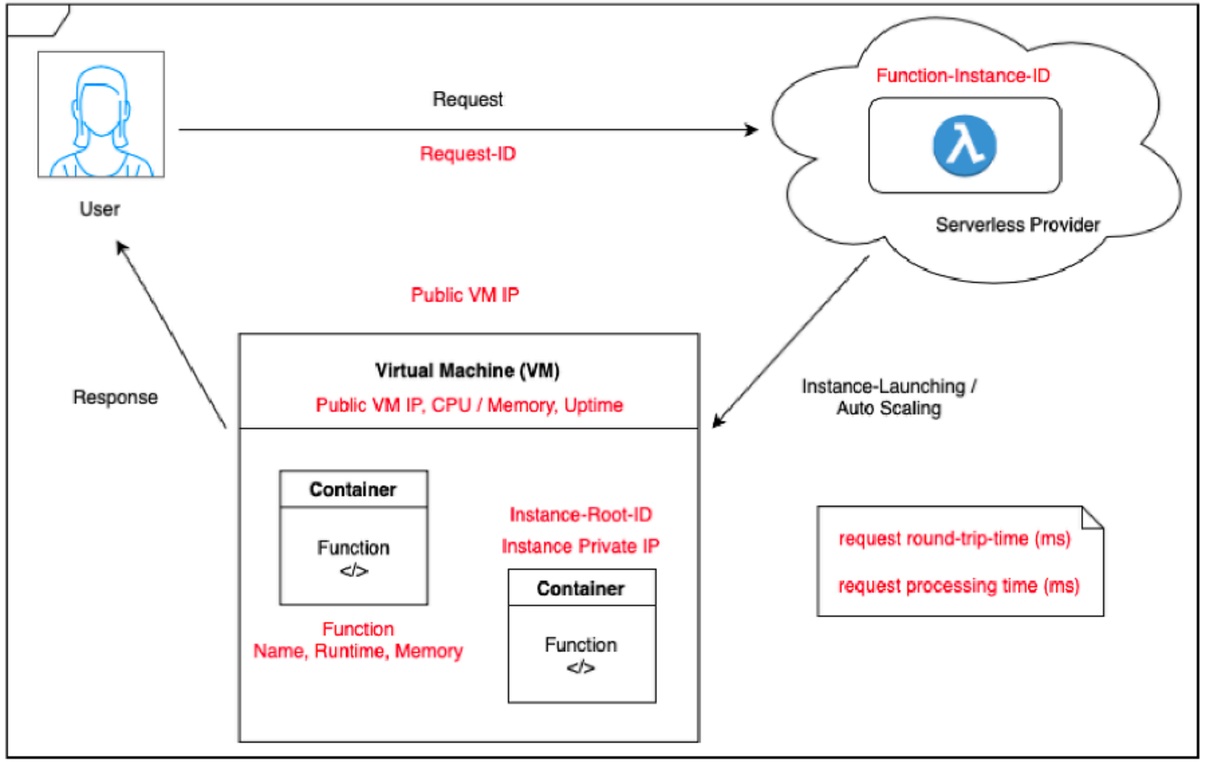
Shared-Disk Parallelism

Modern networks in data centers have become much faster: 100GbE to even TbE!

- **Decoupling** of compute+memory from storage is common in cloud
  - *Hybrids* of shared-disk parallelism + shared-nothing parallelism
  - E.g, store datasets on S3 and read as needed to local EBS

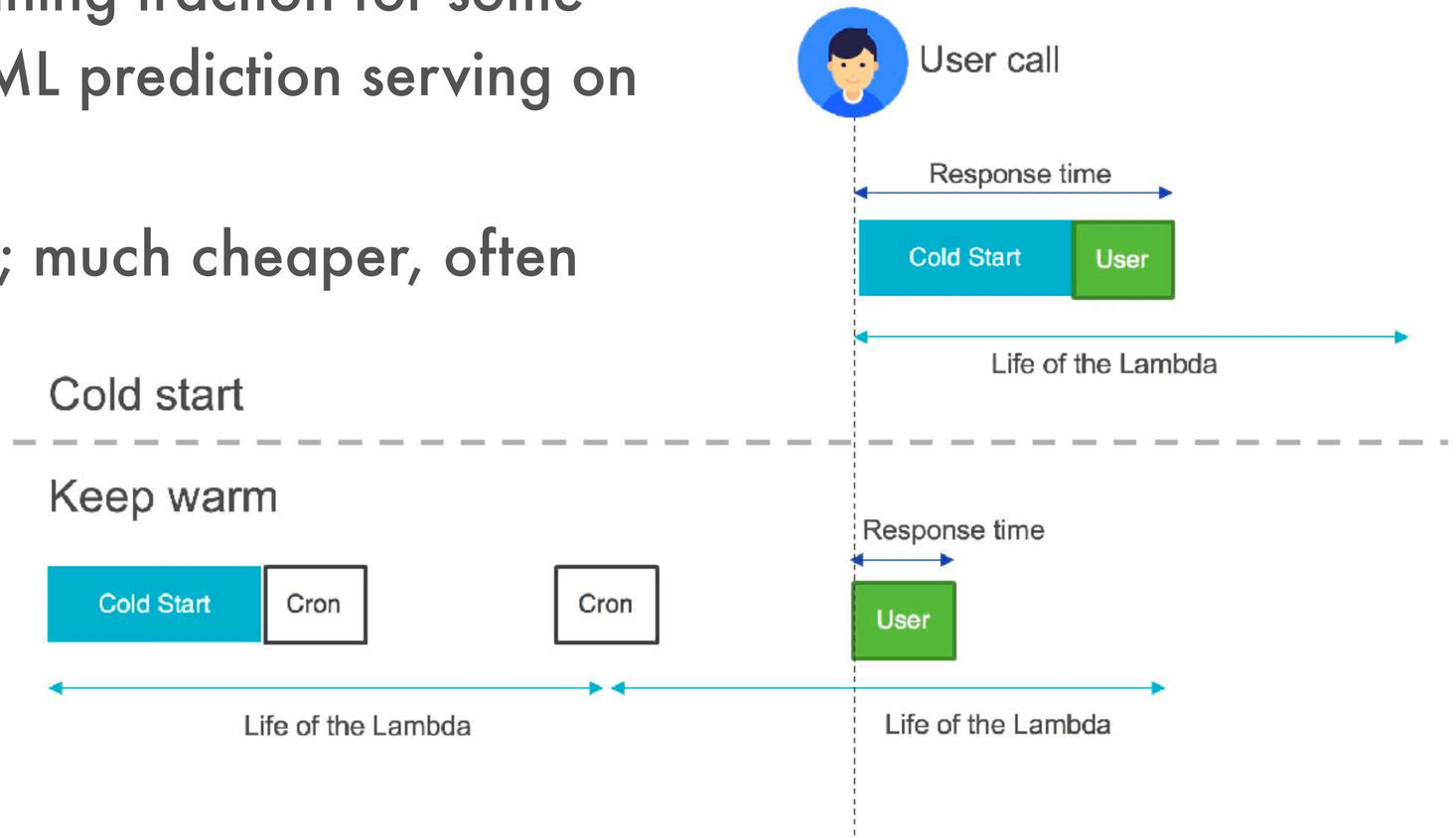
Cloud 3.0 (Ongoing Research)

- Full **resource disaggregation**! That is, compute, memory, storage, etc. are all network-attached and elastically added/removed
- User gives a program (function) to run and specifies CPU and DRAM needed
- Cloud provider abstracts away all resource provisioning entirely
- Aka *Function-as-a-Service* (FaaS)



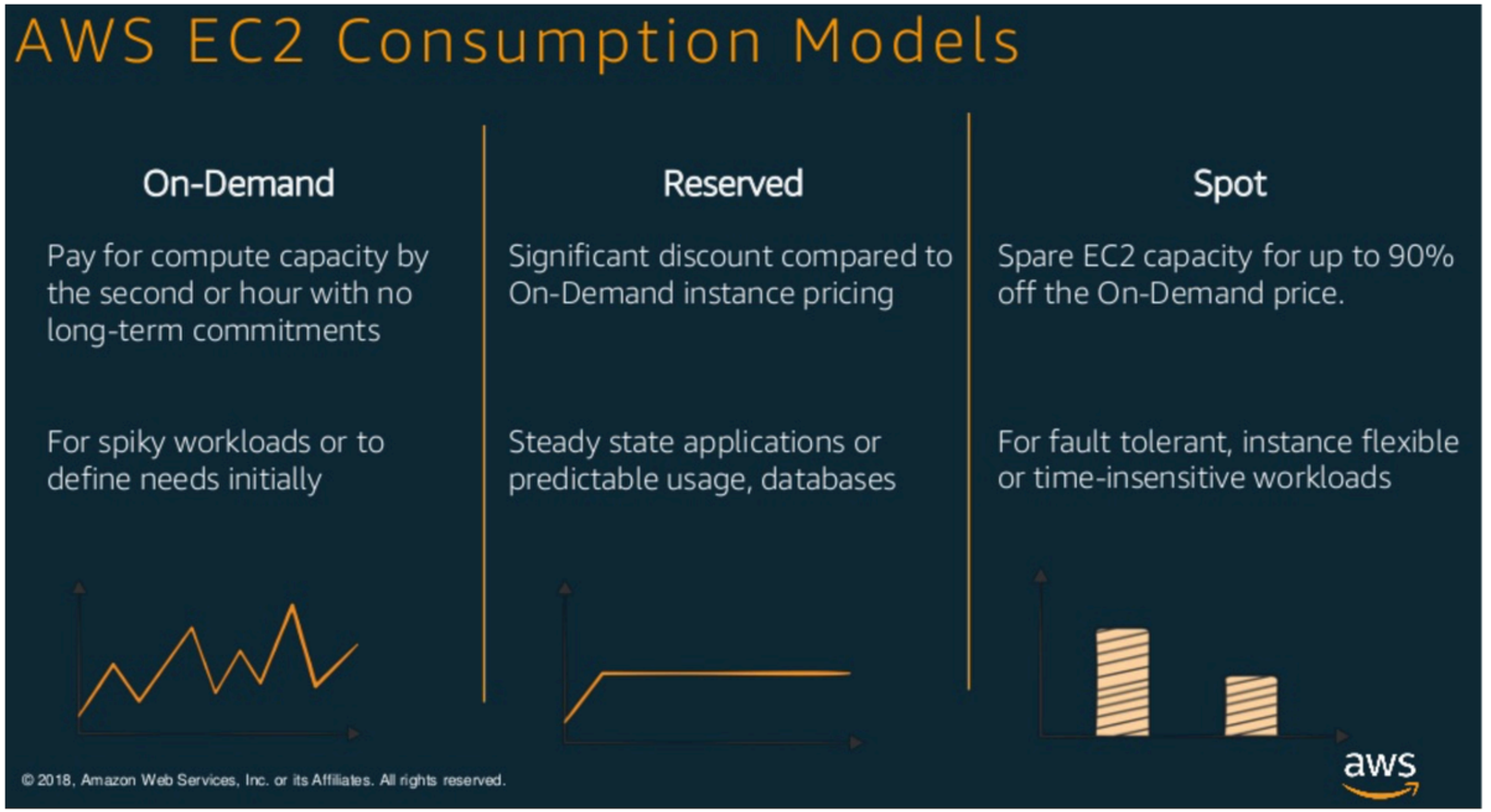
Cloud 3.0 (Ongoing Research)

- "Serverless" and disaggregated resources all connected to fast networks
- **Serverless** paradigm gaining traction for some applications, e.g., online ML prediction serving on websites
- Higher resource efficiency; much cheaper, often by 10x vs Spot instances



New Cloud Renting Paradigms

- Cloud 2.0's flexibility enables radically different paradigms
- AWS example below; Azure and GCP have similar gradations



https://www.slideshare.net/AWSUsersGroupBengalu/amazon-ec2-spot-instances

More on Spot vs On-Demand

	Spot Instances	On-Demand Instances
Launch time	Can only be launched immediately if the Spot Request is active and capacity is available.	Can only be launched immediately if you make a manual launch request and capacity is available.
Available capacity	If capacity is not available, the Spot Request continues to automatically make the launch request until capacity becomes available.	If capacity is not available when you make a launch request, you get an insufficient capacity error (ICE).
Hourly price	The hourly price for Spot Instances varies based on demand.	The hourly price for On-Demand Instances is static.
Rebalance recommendation	The signal that Amazon EC2 emits for a running Spot Instance when the instance is at an elevated risk of interruption.	You determine when an On-Demand Instance is interrupted (stopped, hibernated, or terminated).
Instance interruption	You can stop and start an Amazon EBS-backed Spot Instance. In addition, the Amazon EC2 Spot service can interrupt an individual Spot Instance if capacity is no longer available, the Spot price exceeds your maximum price, or demand for Spot Instances increases.	You determine when an On-Demand Instance is interrupted (stopped, hibernated, or terminated).

https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-spot-instances.html

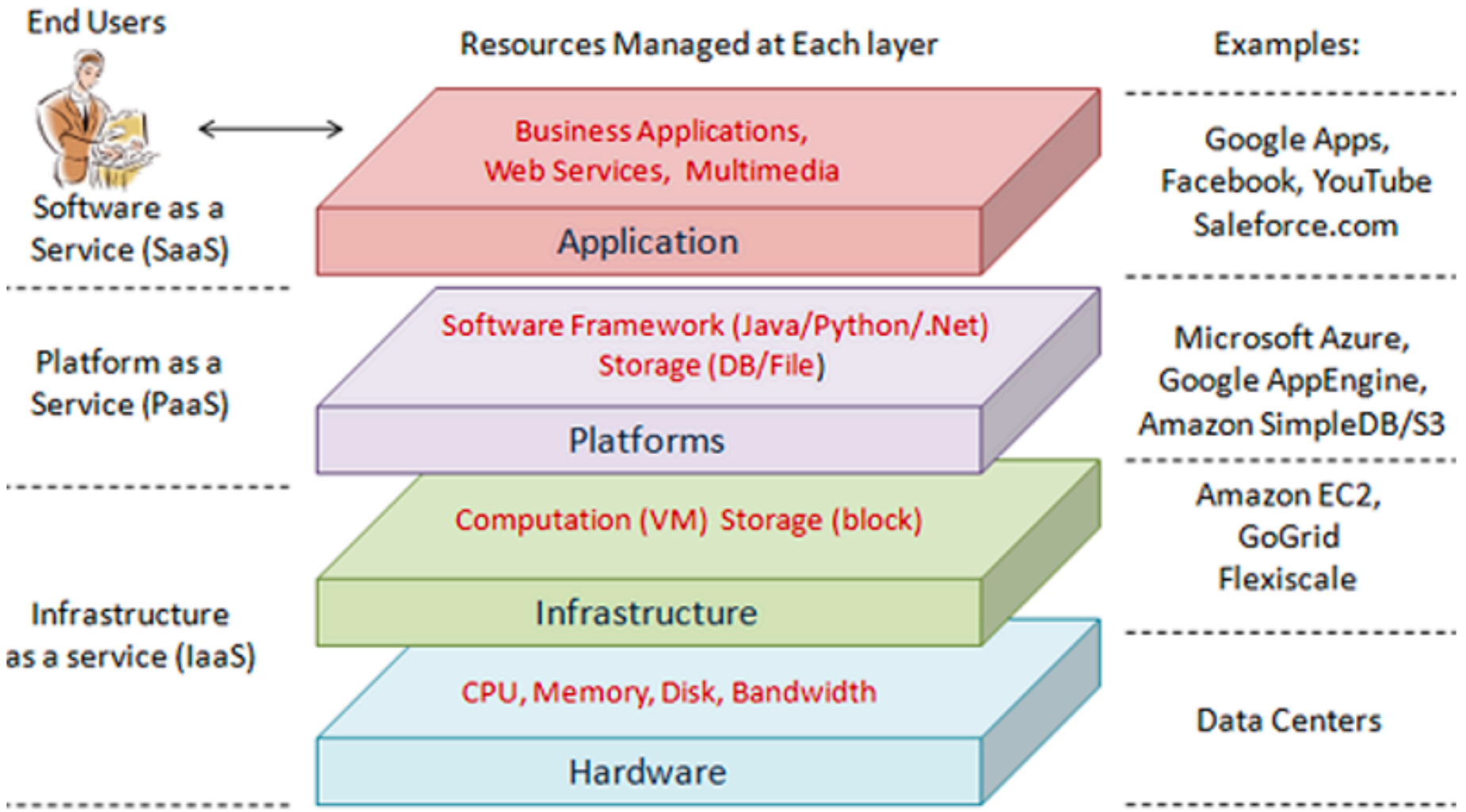
Advantage and disadvantage

- Cloud 1.0:
  - +: Simple, Perfect isolation,
  - - : Expensive.
- Cloud 2.0:
  - +: Cheaper than Cloud 1.0.
  - - : Some resource waste
- Cloud 3.0:
  - +: Cheapest
  - - : Cold-start issues, Security & Privacy, Hard to manage.

Today’s topic

- Cloud computing evolution - sharing granularity
- Cloud computing layers
- Cloud computing practices

Cloud Computing



https://benchpartner.com/layered-model-of-cloud-computing

Analytics

Amazon Athena
Query data in S3 using SQL

Amazon CloudSearch
Managed search service

Amazon DataZone (Preview)
Unlock data across organizational boundaries with built-in governance

Amazon OpenSearch Service
Search, visualize, and analyze up to petabytes of text and unstructured data

Amazon EMR
Easily run big data frameworks

Amazon FinSpace
Analytics for the financial services industry

Amazon Kinesis
Analyze real-time video and data streams

Amazon Managed Streaming for Apache Kafka
Fully managed Apache Kafka service

Amazon Redshift
Fast, simple, cost-effective data warehousing

Amazon QuickSight
Fast business analytics service

AWS Clean Rooms
Match, analyze, and collaborate on datasets–without sharing or revealing underlying data

AWS Data Exchange
Find, subscribe to, and use third-party data in the cloud

AWS Data Pipeline
Orchestration service for periodic, data-driven workflows

AWS Glue
Simple, scalable, and serverless data integration

AWS Lake Formation
Build, manage, and secure your data lake

Resources and Media

Blog
Read the latest blog posts on Big Data

What's New on AWS
See announcements for Big Data and Analytics on AWS

Customer Enablement

AWS IQ
Complete your projects faster with help from AWS Certified third-party experts

AWS Managed Services
Operate your AWS infrastructure for you

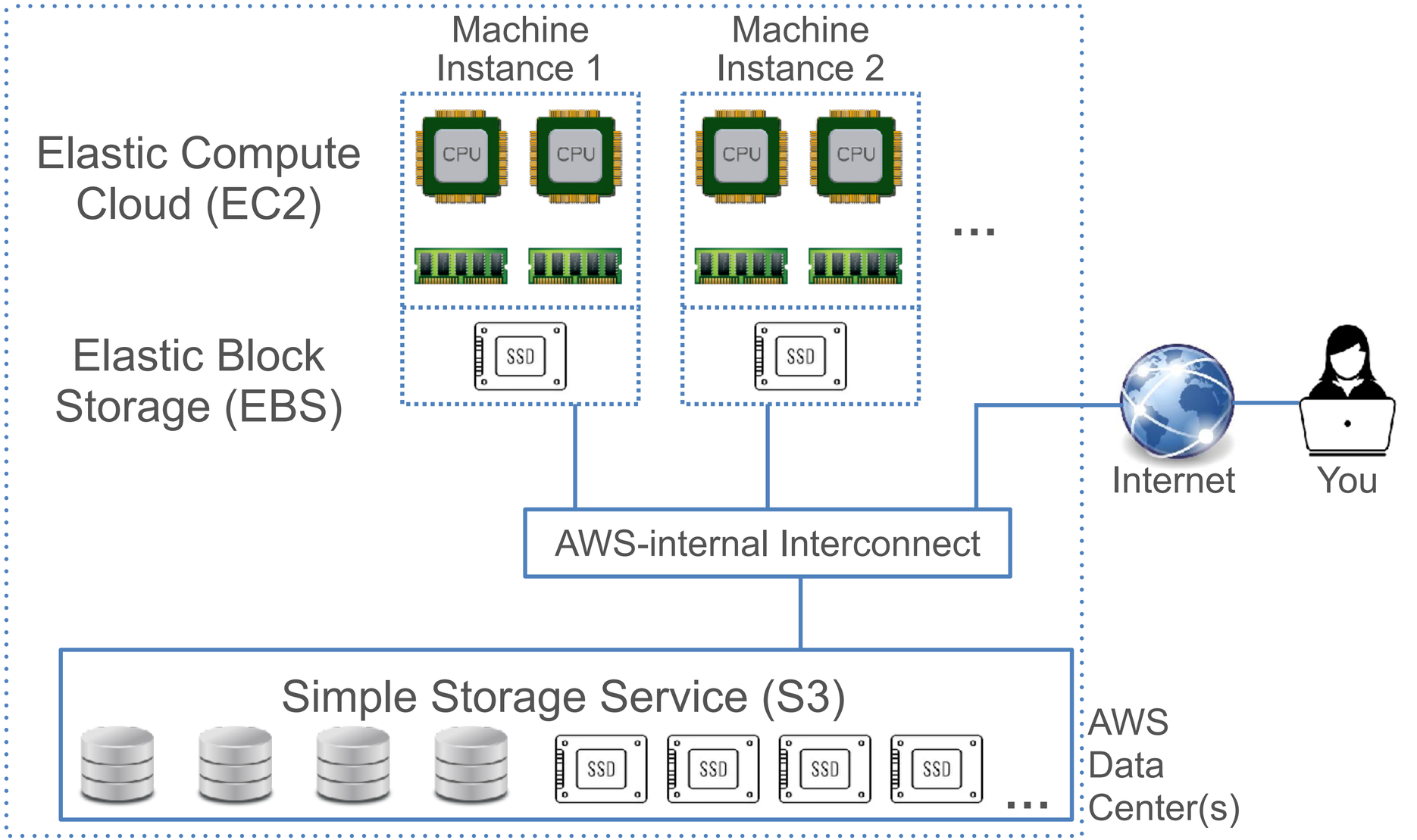
AWS Professional Services
Accelerate your business outcomes with AWS

AWS Training and Certification
Build and validate your skills and technical expertise

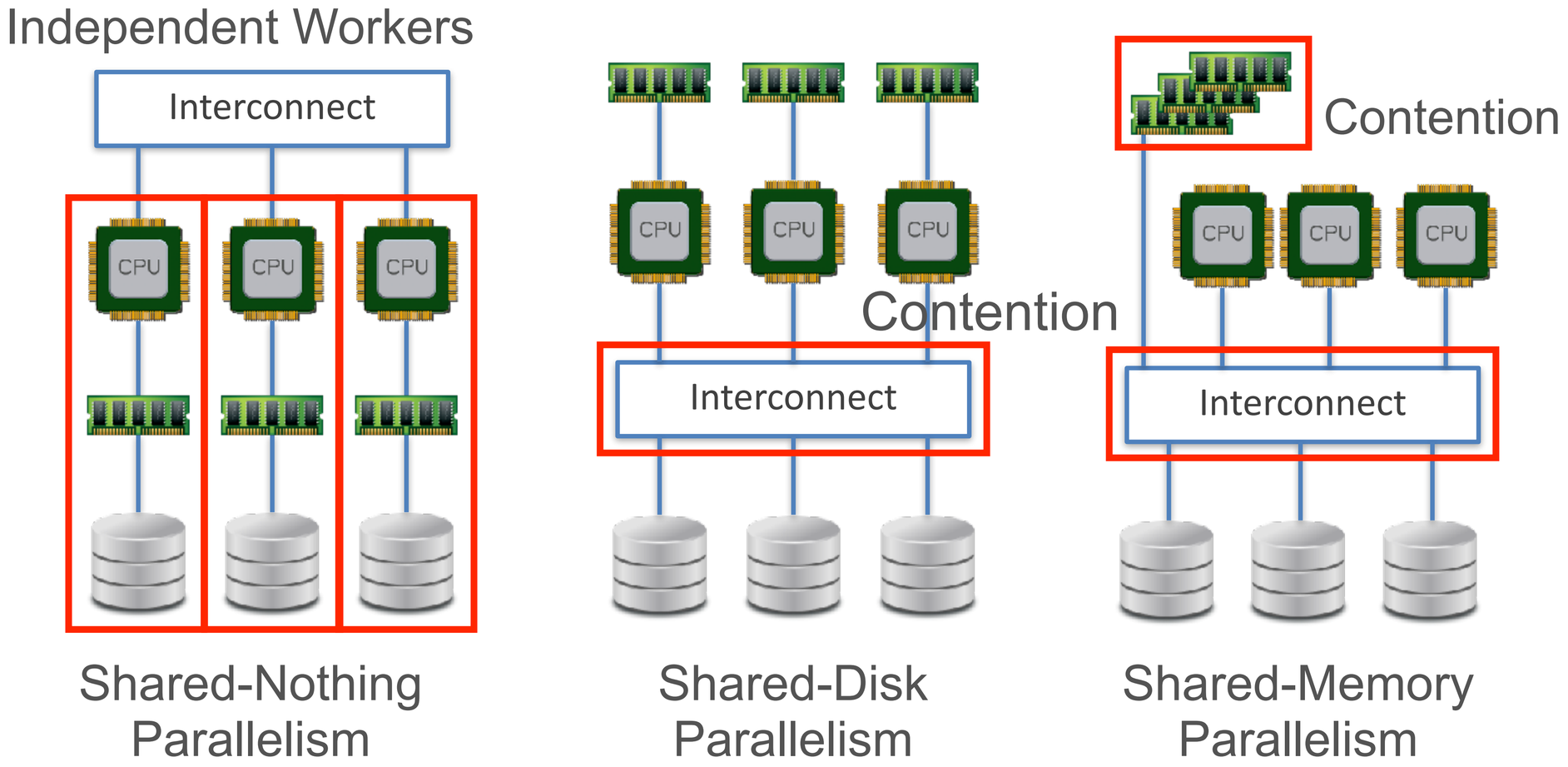
Examples of AWS Cloud Services

- **IaaS**:
  - **Compute**: EC2, ECS, Fargate, Lambda
  - **Storage**: S3, EBS, EFS, Glacier
  - **Networking**: CloudFront, VPC
- **PaaS**:
  - **Database/Analytics Systems**: Aurora, Redshift, Neptune, ElastiCache, DynamoDB, Timestream, EMR, Athena
  - **Blockchain**: QLDB; **IoT**: Greengrass
- **SaaS**:
  - **ML/AI**: SageMaker, Elastic Inference, Lex, Polly, Translate, Transcribe, Textract, Rekognition, Ground Truth
  - **Business Apps**: Chime, WorkDocs, WorkMail

Example: AWS Services for PA1



3 Paradigms of Multi-Node Parallelism

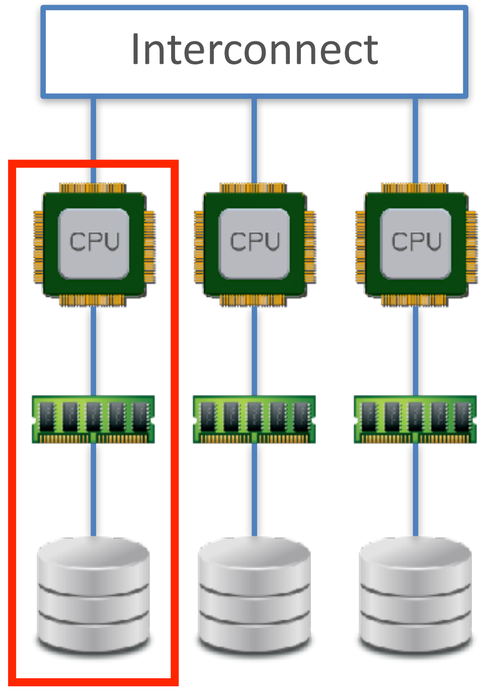


Most parallel RDBMSs (Teradata, Greenplum, Redshift), Hadoop, and Spark use shared-nothing parallelism

New Cloud Renting Paradigms

Such bundling means some applications might under-utilize some resources!

- **Serverless** paradigm gaining traction for some applications, e.g., online ML prediction serving on websites
- User gives a program (function) to run and specifies CPU and DRAM needed
- Cloud provider abstracts away all resource provisioning entirely
- Higher resource efficiency; much cheaper, often by 10x vs Spot instances
- Aka *Function-as-a-Service* (FaaS)



Shared-Nothing Parallelism

Today’s topic

- Cloud computing evolution - sharing granularity
- Cloud computing layers
- Advantages of Cloud computing

OMG, is all this complexity worth it?!

- Depends on user's/application's Pareto tradeoffs! :)
- **On-premise** cluster are still common in large enterprises, healthcare, and academia; "hybrid clouds" too
- Recall main pros of cloud: manageability, cost, and elasticity
- Some main cons of cloud (vs on-premise):
  - **Complexity** of composing cloud APIs and licenses; data scientists must keep relearning; "CloudOps" teams
  - **Cost** over time can *crossover* and make it costlier!
  - Easier to **waste money** accidentally on the fly
  - **"Lock-in"** by cloud vendor
  - **Privacy**, **security**, and **governance** concerns
  - **Internet disruption** or **unplanned downtime**, e.g., AWS outage in 2015 made Netflix, Tinder, etc. unavailable! :)

Recap: Cloud Computing v.s. on-premise clusters

- Compute, storage, memory, networking, etc. are virtualized and exist on *remote servers*; *rented* by application users
- Main pros of cloud vs on-premise clusters:
  - **Manageability**: Managing hardware is not user's problem
  - **Pay-as-you-go**: Fine-grained pricing economics based on actual usage (granularity: seconds to years!)
  - **Elasticity**: Can dynamically add or reduce capacity based on actual workload's demand
- Infrastructure-as-a-Service (IaaS); Platform-as-a-Service (PaaS); Software-as-a-Service (SaaS)

Review Questions

1. What are the 3 main layers of a typical cloud? Give examples of AWS services in each layer. Which ones do your PAs use?
2. What is a benefit of separating PaaS from SaaS in cloud?
3. Briefly explain 1 pro and 1 con of Shared Disk Parallelism vs Shared Nothing Parallelism.
4. Briefly explain 1 pro and 1 con of On-Demand vs Spot instances on AWS.
5. What is so great about the serverless cloud anyway?
6. What is so great about resource disaggregation in future clouds?
7. Briefly explain 2 pros and 2 cons of cloud vs on-premise clusters.