AI-Systems
Machine Learning in the Cloud

Joseph E. Gonzalez
Co-director of the RISE Lab
jegonzal@cs.berkeley.edu

What is cloud computing?

"The interesting thing about Cloud Computing is that we've redefined Cloud Computing to include everything that we already do. . . . I don't understand what we would do differently in the light of Cloud Computing other than change the wording of some of our ads."

-- **Larry Ellison**,
Wall Street Journal, 2008

Quote from "Above the Clouds: A Berkeley View of Cloud Computing"

8 years later …

2016

"If 'cloud computing' has a meaning, it is not a way of doing computing, but rather a way of thinking about computing: a devil-may-care approach which says, 'Don't ask questions. Don't worry about who controls your computing or who holds your data. Don't check for a hook hidden inside our service before you swallow it. Trust companies without hesitation.' In other words, 'Be a sucker.' "

-- **Richard Stallman,**
Boston Review, 2010

https://bostonreview.net/articles/richard-stallman-free-software-drm/

RAD Lab
2006-2011

Early paper on cloud computing and helped drive **excitement in the field**.

Made the case for cloud computing
- Illusion of infinite resources
- Elimination of up-front costs
- Pay-per-use
- Economies of scale for everyone
- Simplified operations
- Higher hardware utilization

Above the Clouds: A Berkeley View of Cloud Computing

Published in 2009 and received over **8,4000 citations**

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

Defining Characteristics of the Cloud

- The illusion of infinite computing resources available on demand
- The elimination of an up-front commitment by users
- The ability to pay for use of computing resources on a short-term basis as needed

→ Essentially **Utility Computing**

Public Cloud: when these services are available to the general public

Private Cloud: when these services are sold within a business

Quote from "Above the Clouds: A Berkeley View of Cloud Computing"

The Beginning of the Cloud

1961: John McCarthy presents idea of Utility Computing

2006: Amazon Web Services (AWS) Launched

2008: Google Cloud Launched

2008: Azure Cloud Launched

2009: Berkeley writes: "Above the Clouds: A Berkeley View of Cloud Computing"

Utility Computing is an old idea

"If computers of the kind I have advocated become the computers of the future, then computing may someday be organized ***as a public utility*** *just as the telephone system is a public utility... The computer utility could become the basis of* ***a new and important industry****."*

– John McCarthy

Early Pioneer in **Artificial Intelligence**
Turing award for **contributions to AI**

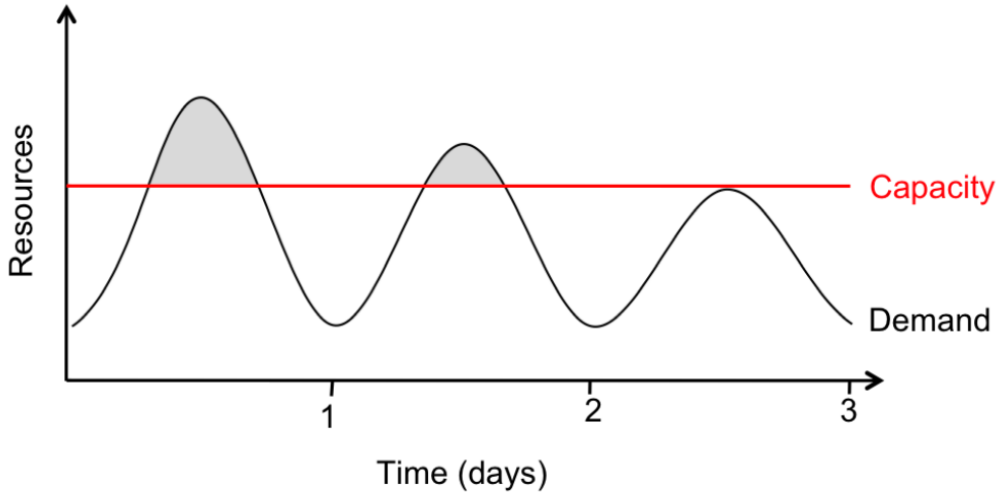
Public Cloud
(>$300B)

Why did it take until 2006?

- **Change is business models** for computing
  - Web2.0 – low commitment, self-serve, pay-as-you-go -- not enough capital to own equipment.
- Existing big-tech companies were aggressively exploiting **economies of scale** to drive down costs
  - Renting becomes cheaper than buying machines
- New **workloads emerged** to leverage **elasticity:**
  - Web applications needed to handle **demand surges**
  - Data intensive apps leveraged scale -- **fast is cheap**

Economics of the Cloud

- **CapEx to OpEx**: transition from large up-front capital expenditures to operational expenditures
  - More money to spend on your launching your business
- **Improved Utilization** through **statistical multiplexing**
  - real-world server utilization for a single business ~ **5% to 20%.**
- **Economies of scales**
  - Negotiate lower hardware prices
  - Spread management costs
  - Leverage existing investments

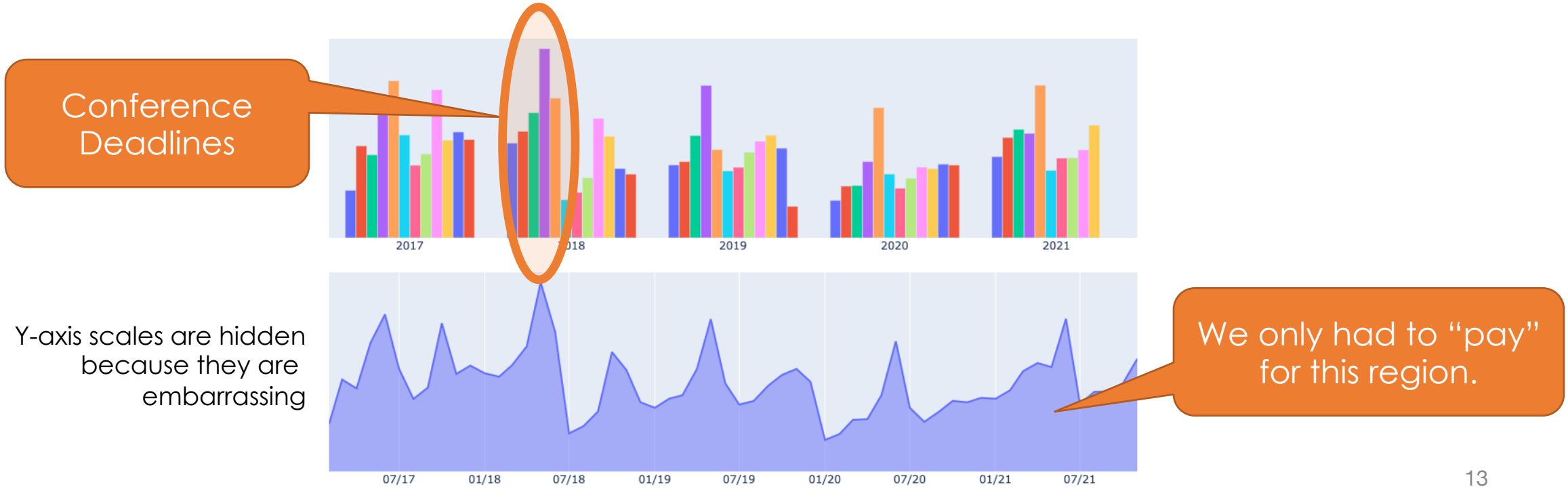


The Cloud Enabled Academic Research

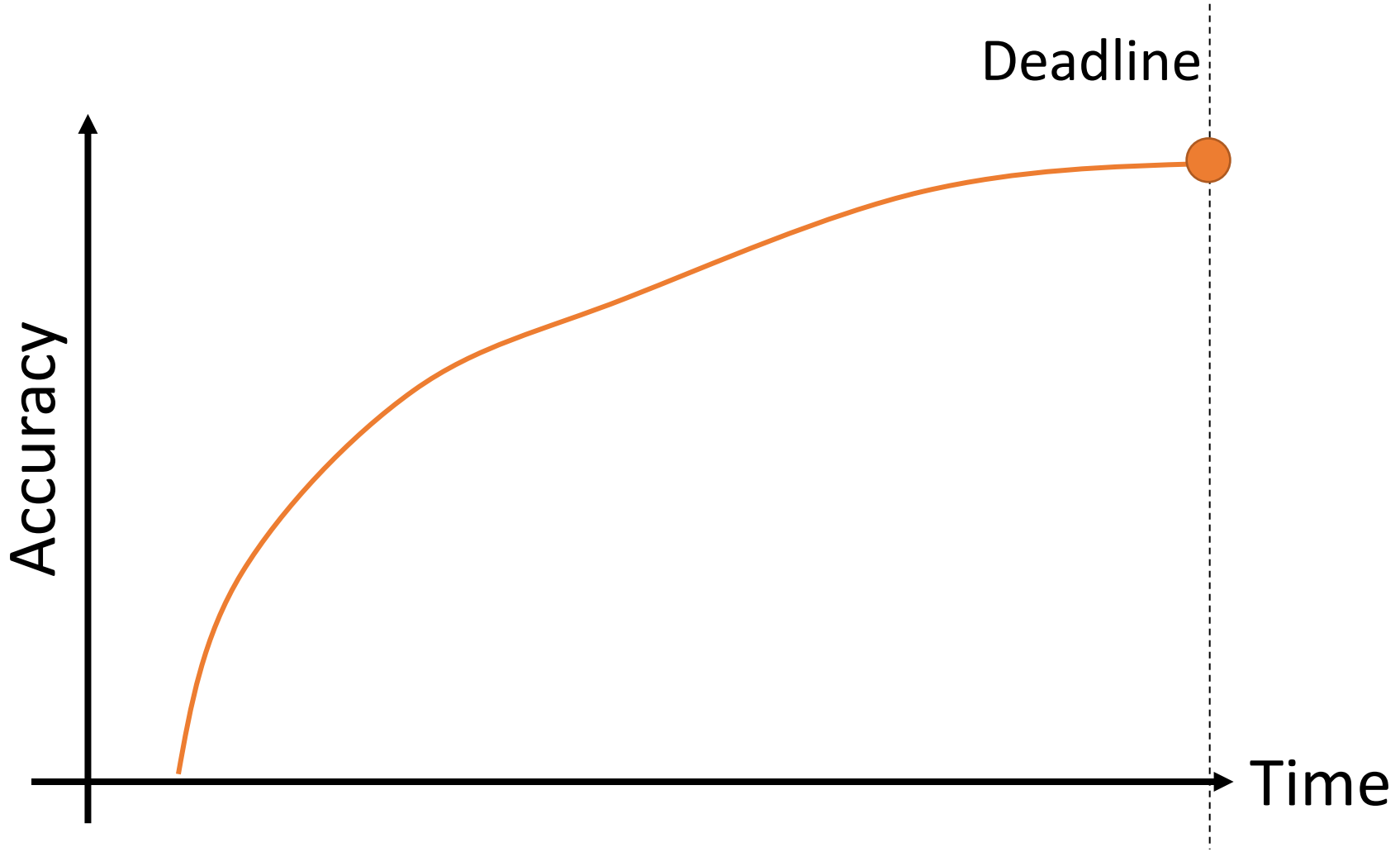
- Access to the **latest hardware**
- Ability to **burst experiments** near conference deadlines
  - Usually…
- Ability for students to build and evaluate **large-scale systems**
  - I would frequently run **concurrent experiments** with **hundreds of machines each**!
- industrial **adoption**
  - Companies can **evaluate open-source (academic) big data tools** without **big upfront investment in hardware.**

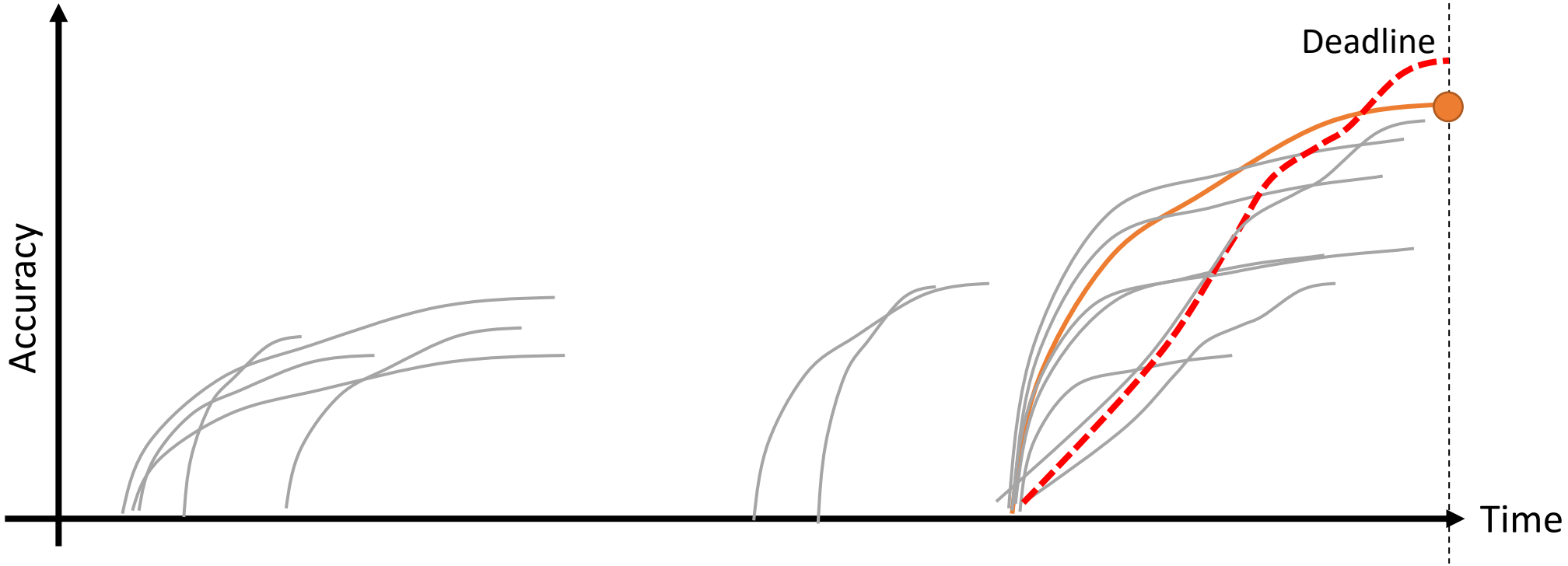
What about the Cloud?

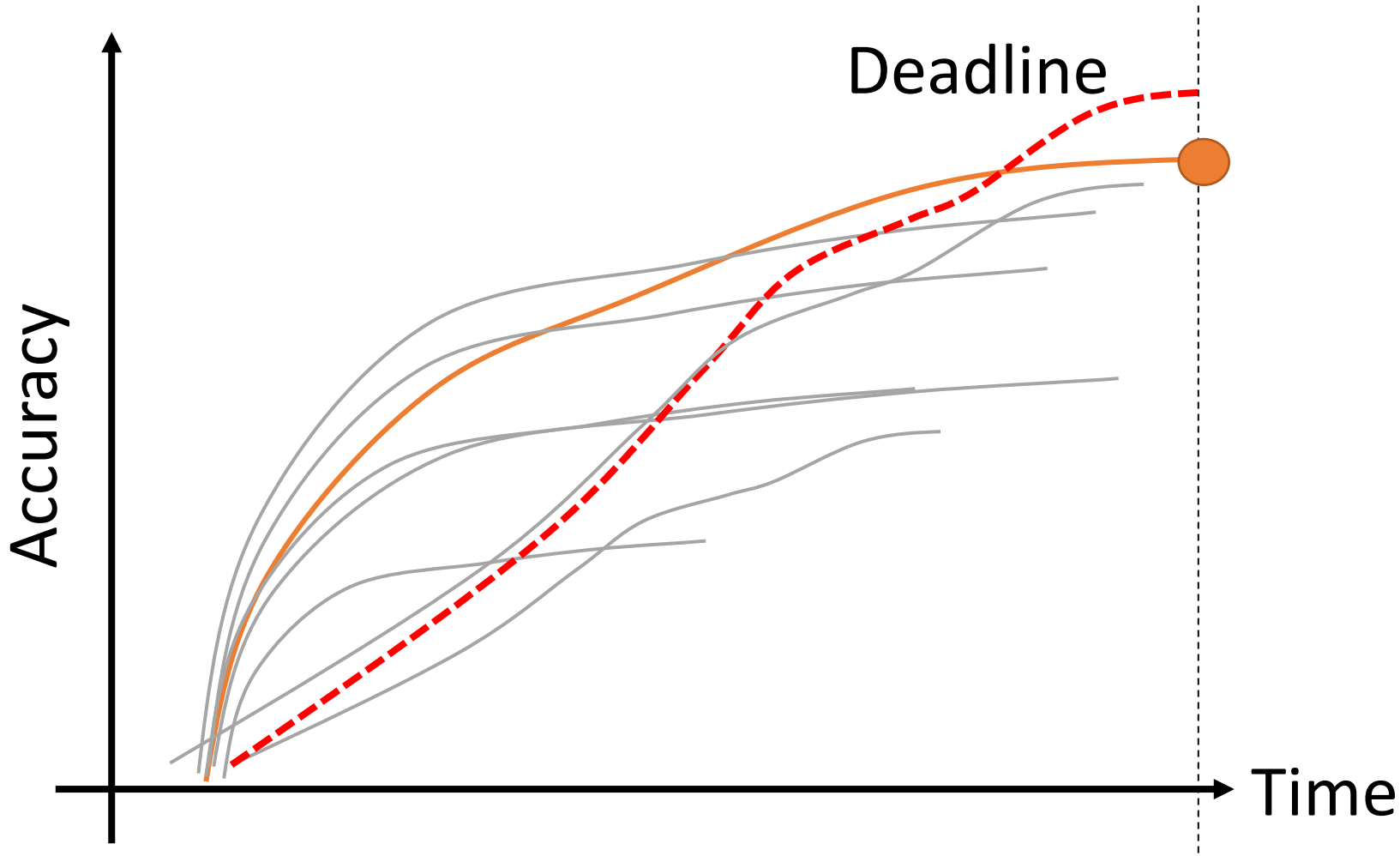
- Access to **latest GPUs** and **TPUs** drove AI research
  - used a LOT OF CREDITS (*thank you AWS, Azure, & Google!*)

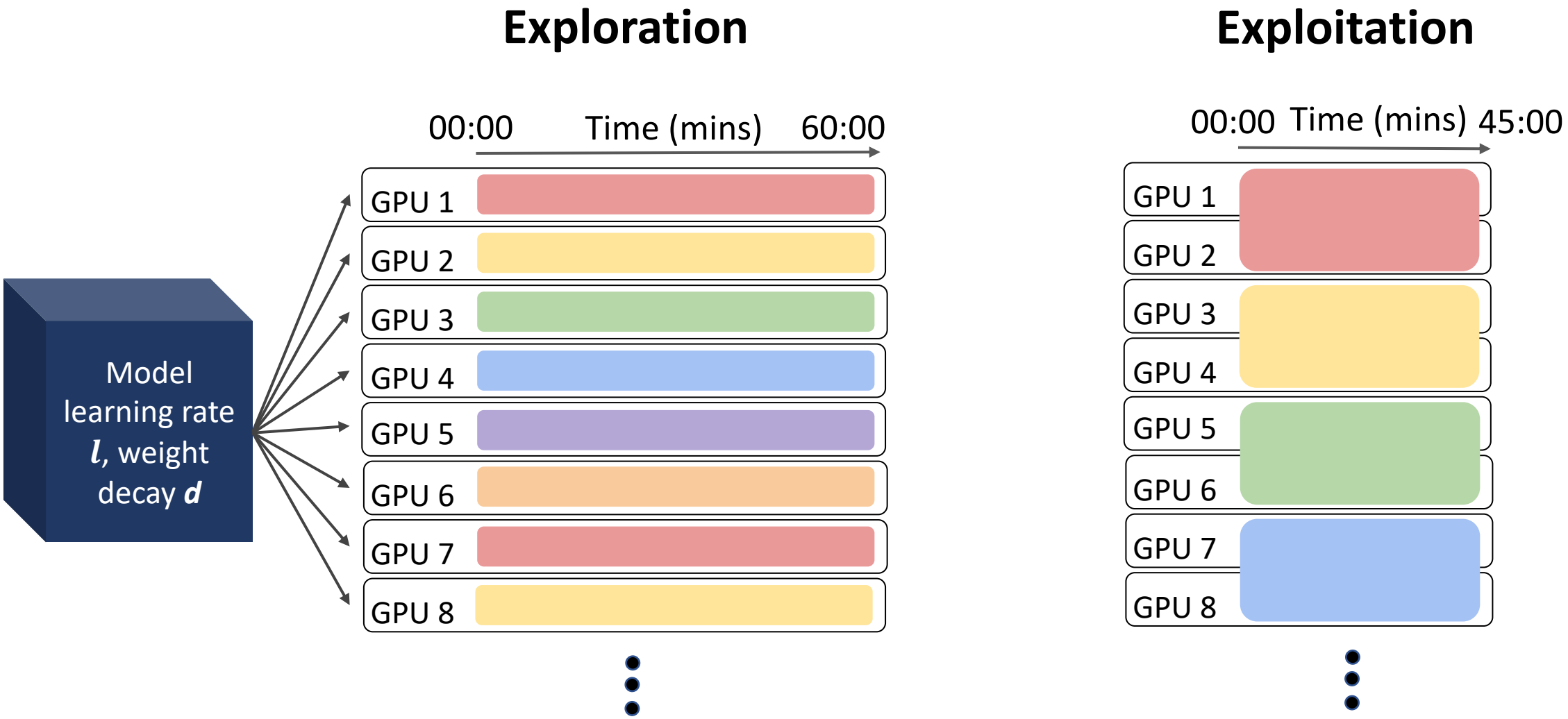


The Elasticity of the cloud drove us to rethink our approach to AI Research

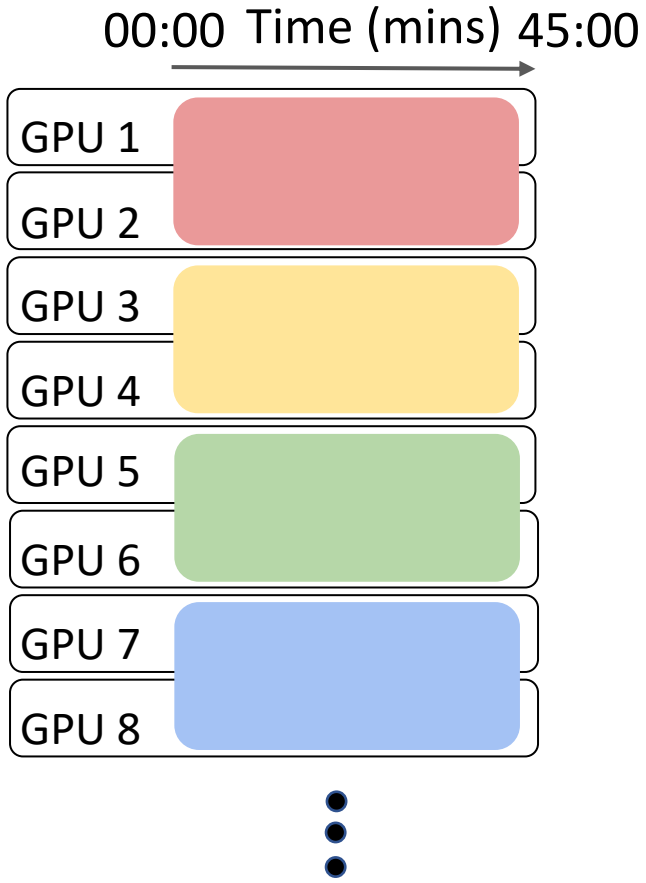

Deadline
Accuracy
Time

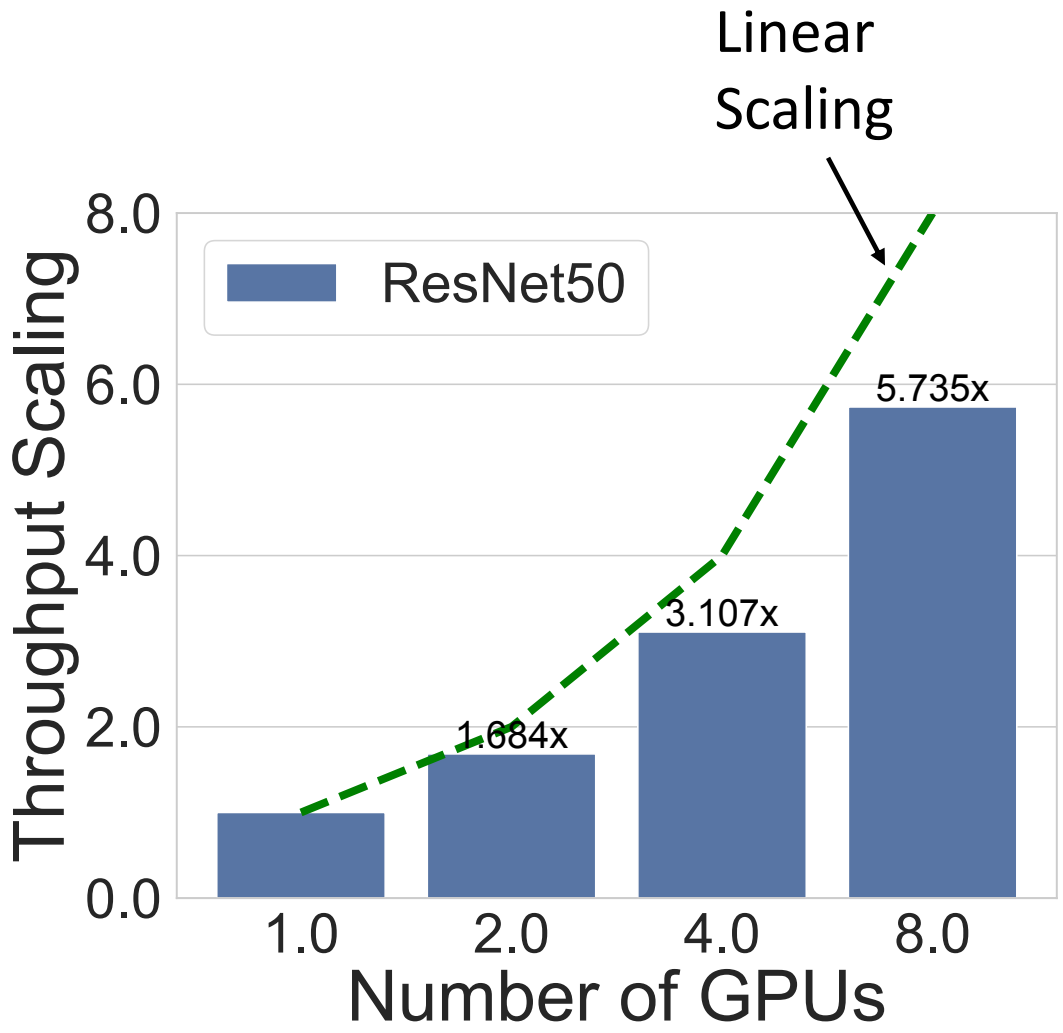

Deadline
Accuracy
Time


Deadline
Accuracy
Time

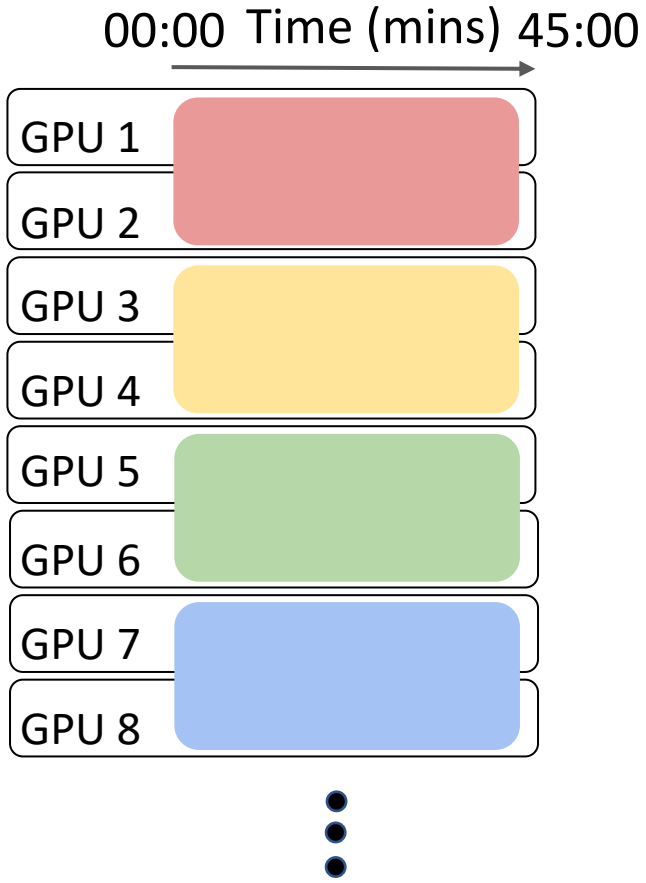

Exploration
00:00
Time (mins)
60:00
Model learning rate l, weight decay d
GPU 1
GPU 2
GPU 3
GPU 4
GPU 5
GPU 6
GPU 7
GPU 8
Exploitation
00:00
Time (mins)
45:00
GPU 1
GPU 2
GPU 3
GPU 4
GPU 5
GPU 6
GPU 7
GPU 8

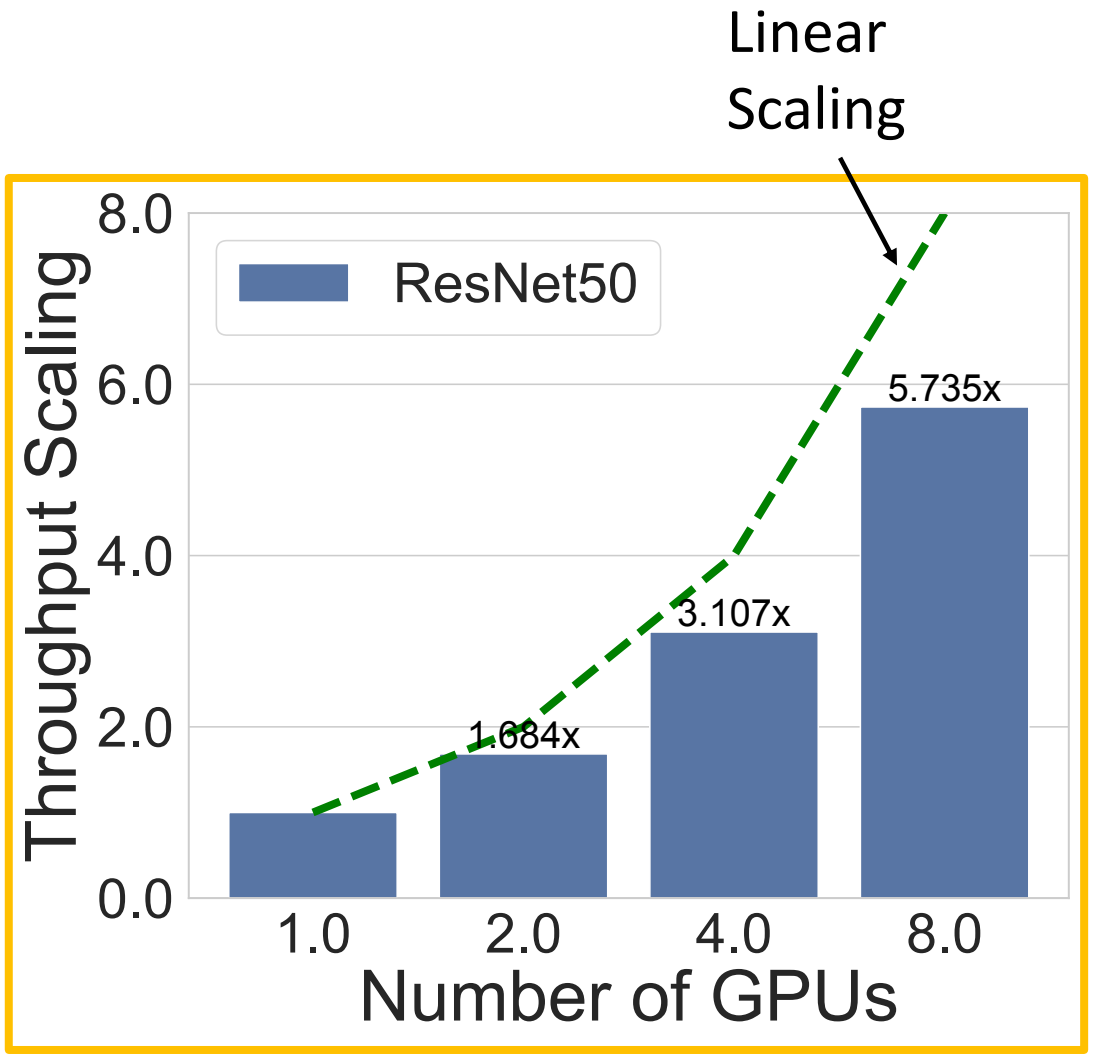
Exploitation





Exploitation





Fixed Cluster

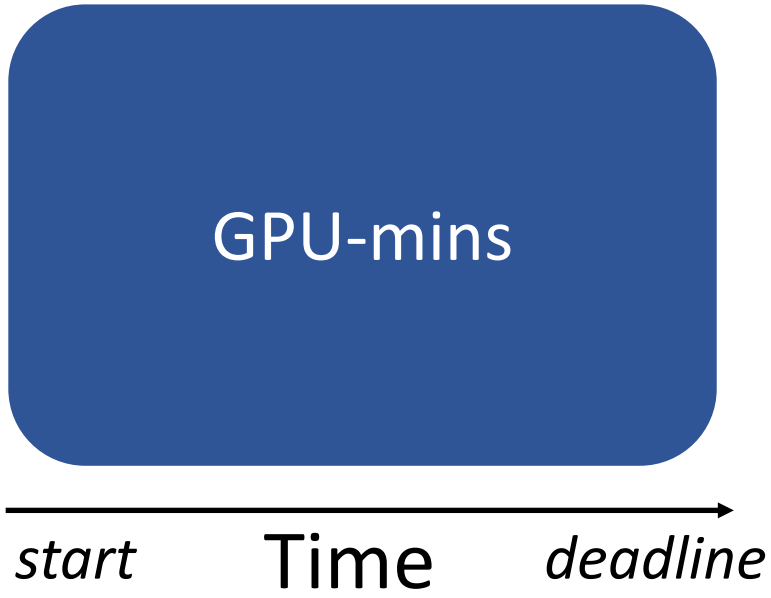
Resources = Machines

Cloud

Resources = Money

Fixed Cluster

Resources = Machines

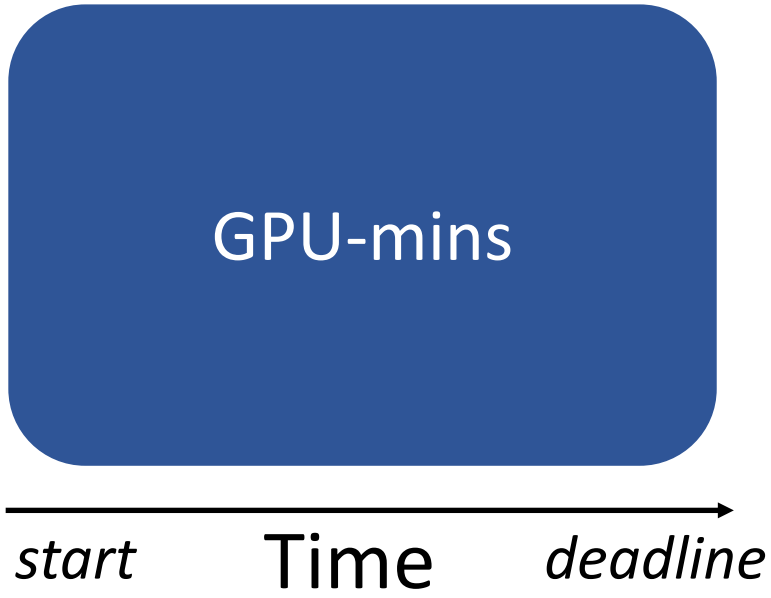


Cloud

Resources = Money

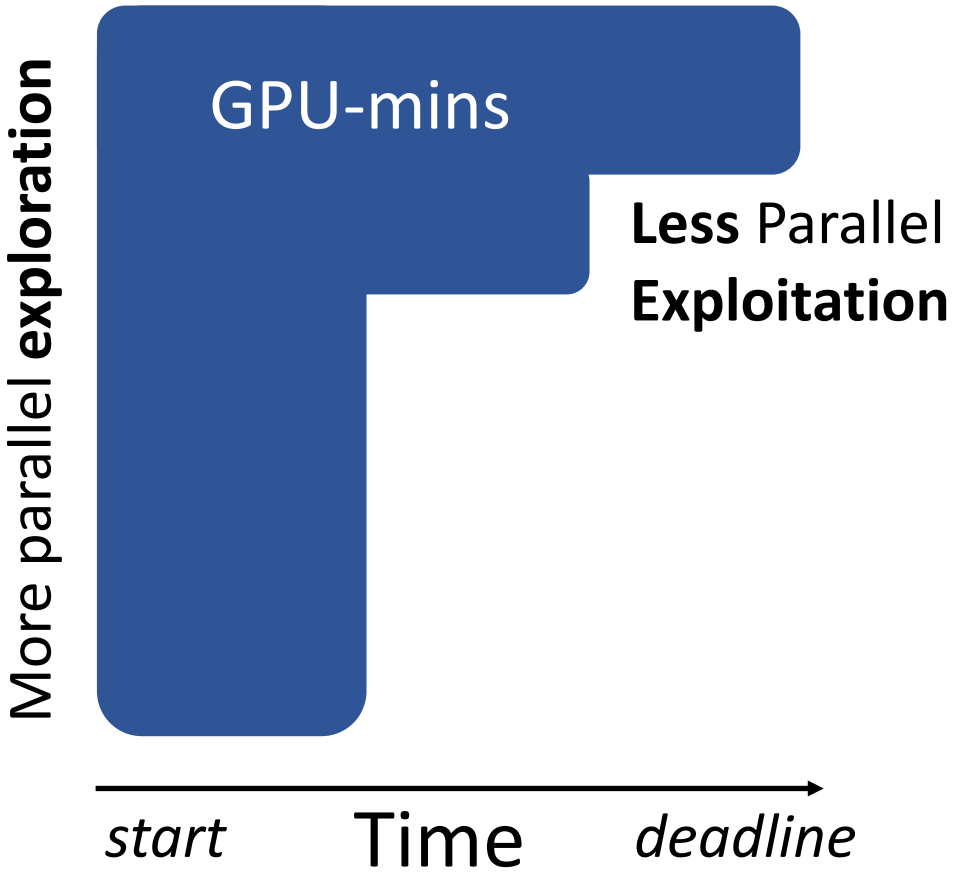
Fixed Cluster

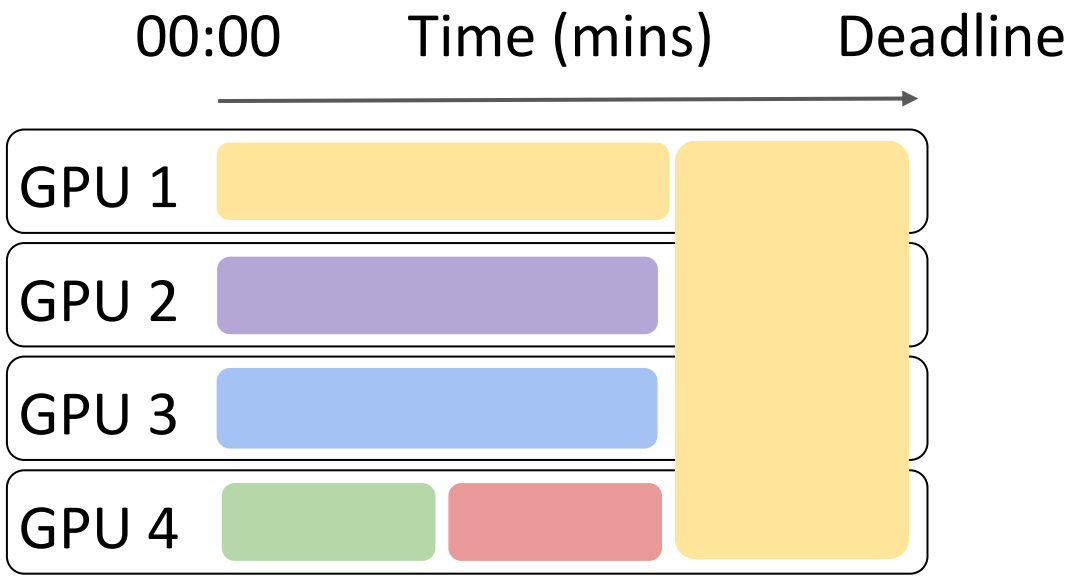
Resources = Machines



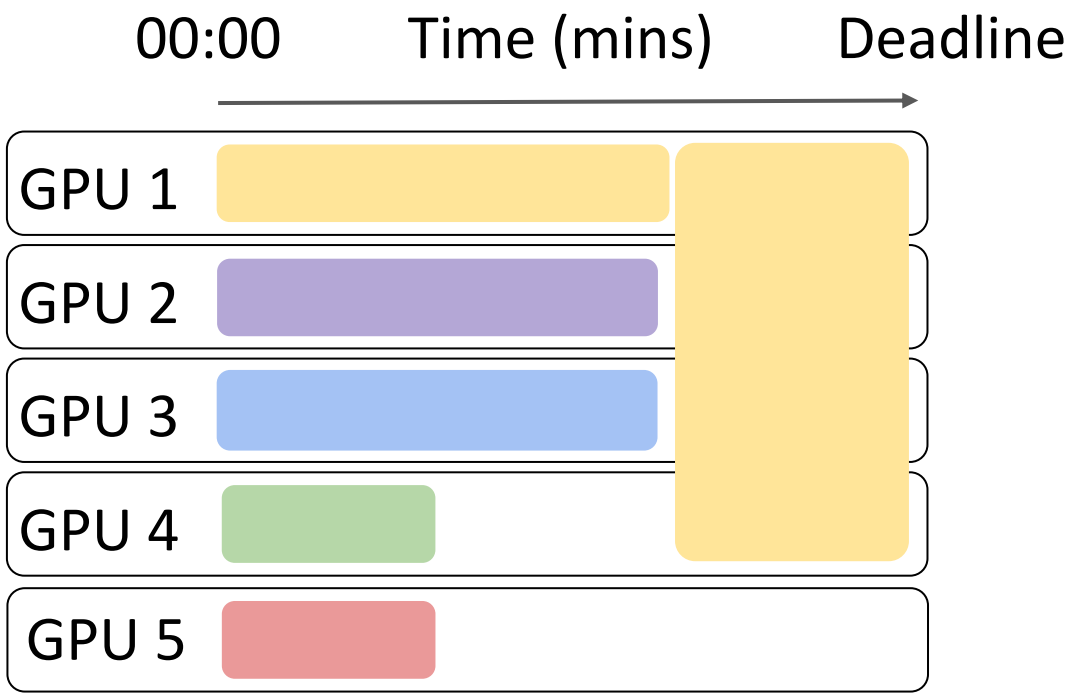
Cloud

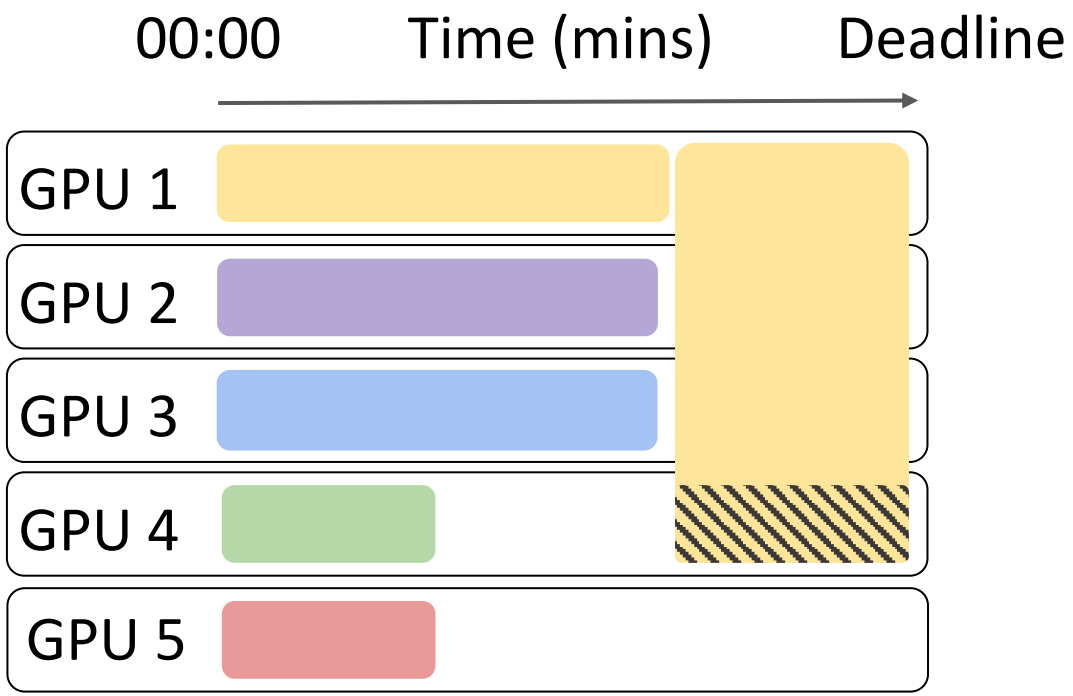
Resources = Money

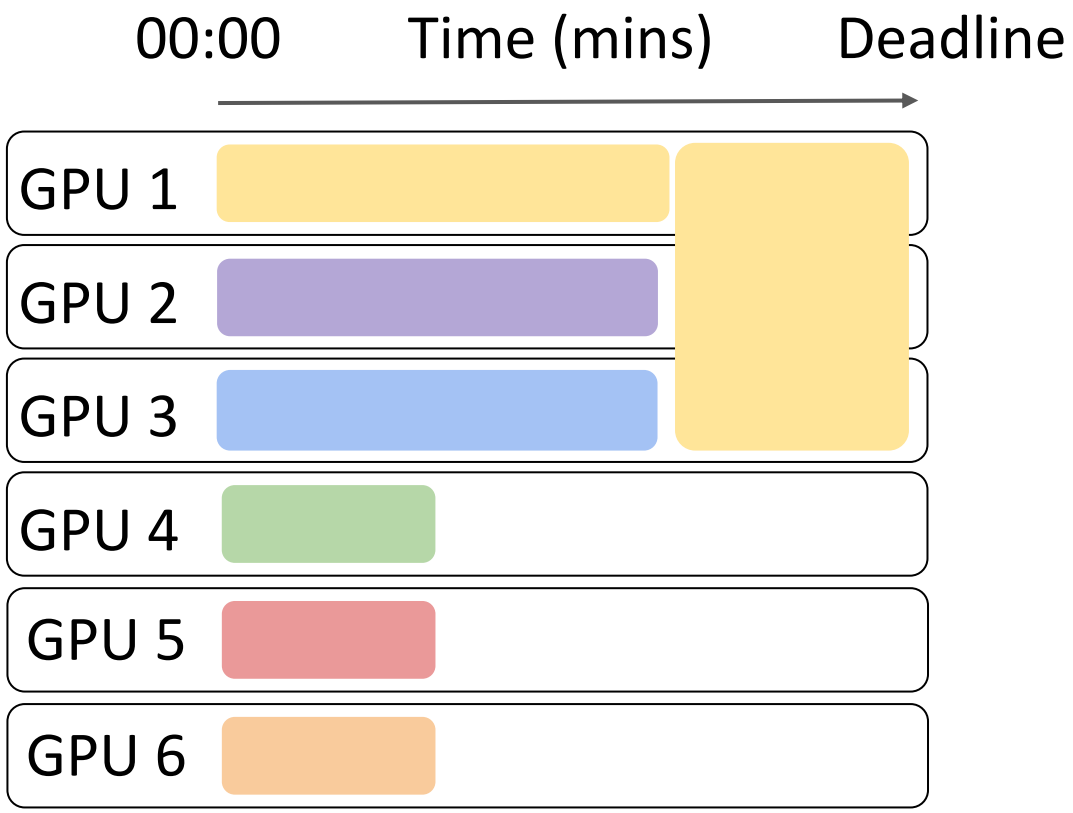


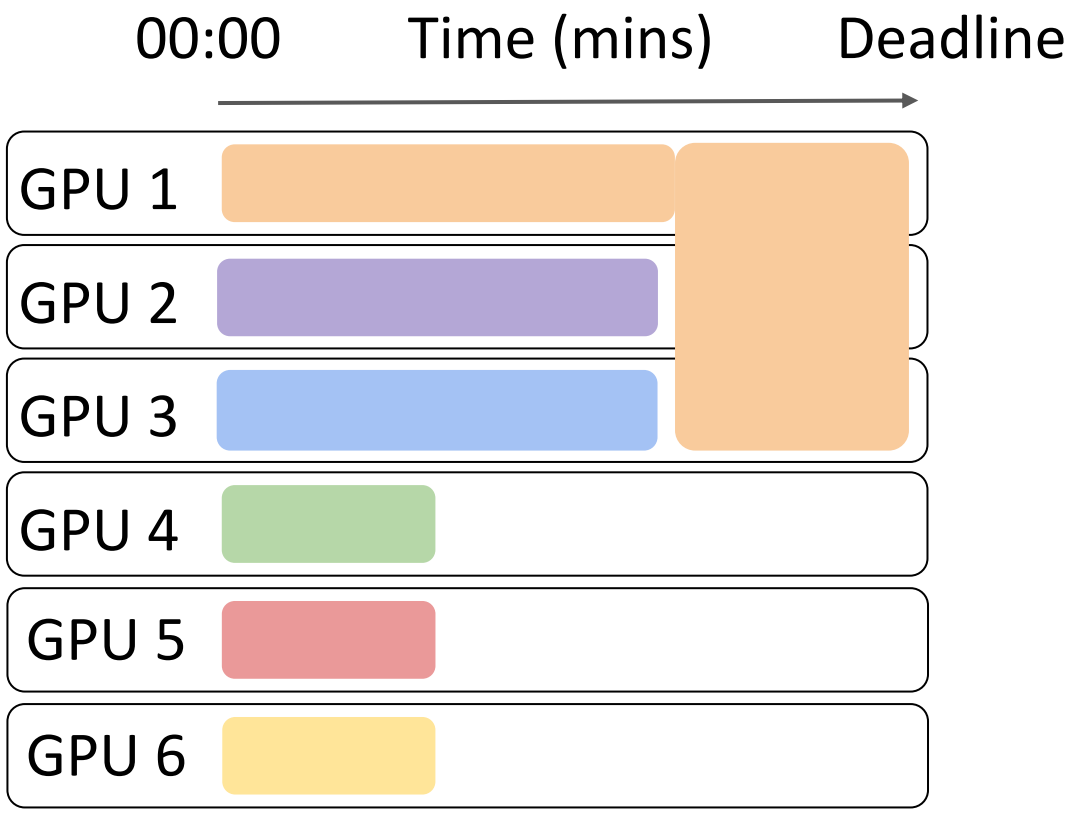


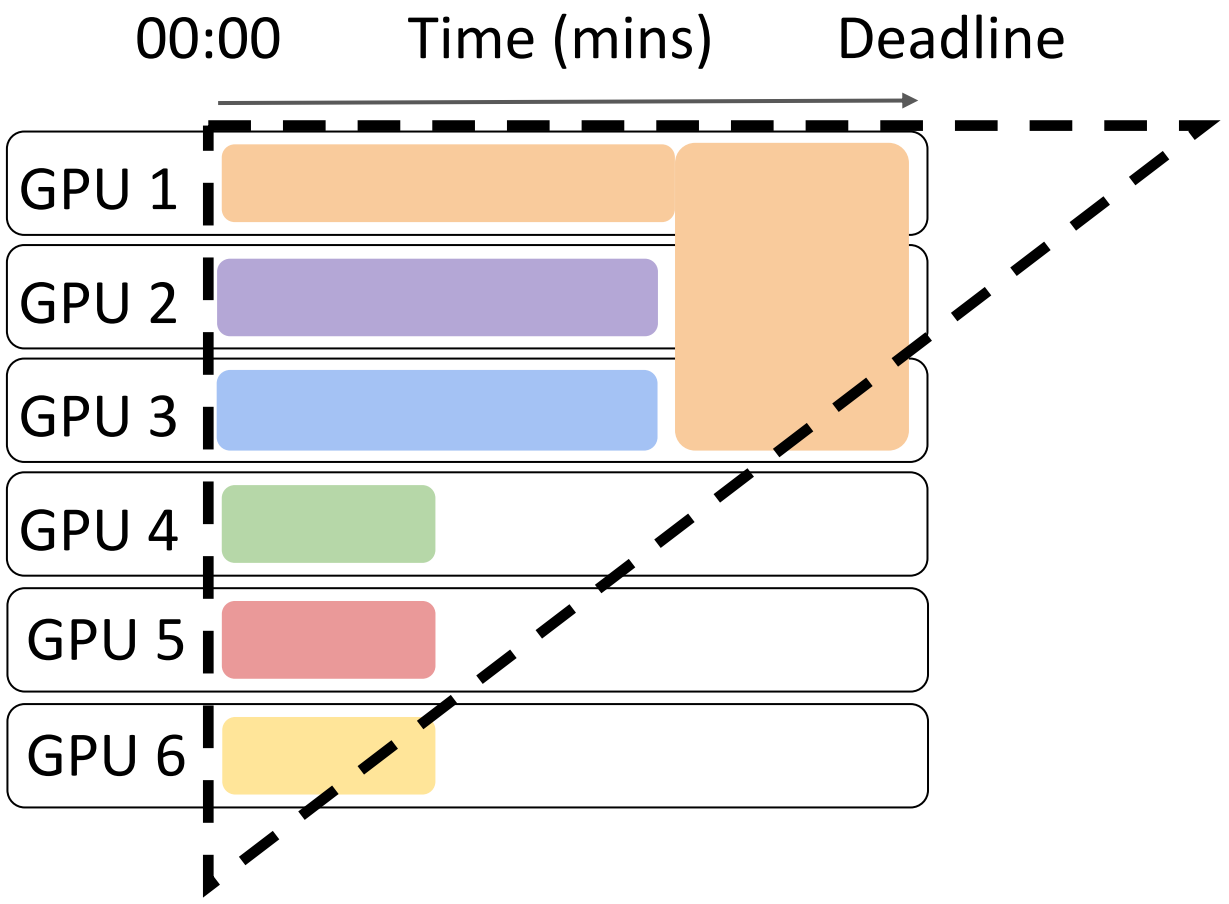
Liaw et al.


00:00
Time (mins)
Deadline
GPU 1
GPU 2
GPU 3
GPU 4
GPU 5

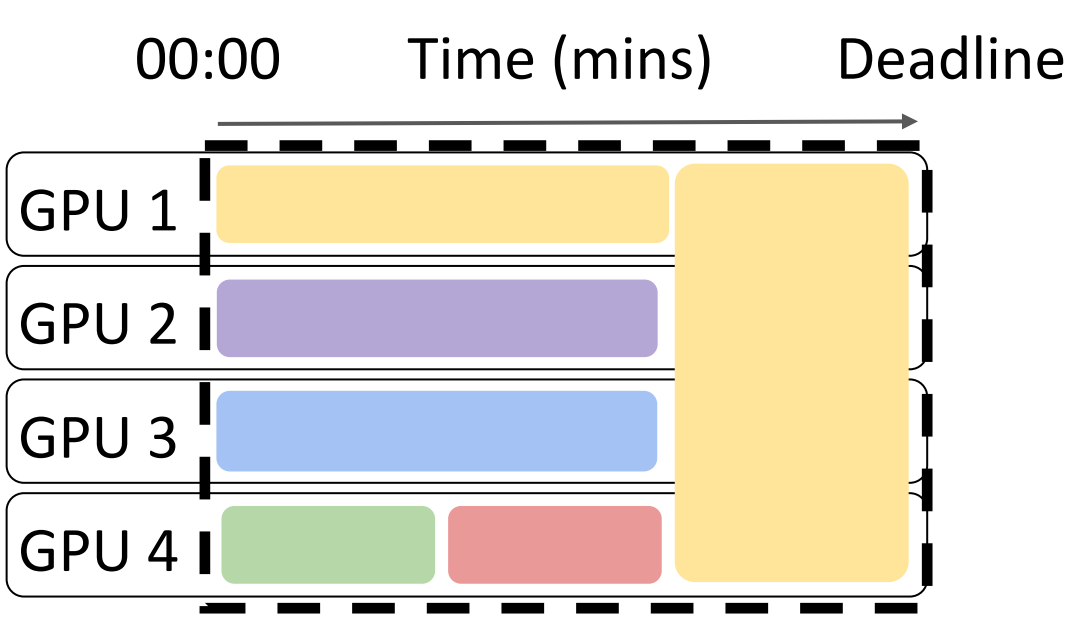

00:00
Time (mins)
Deadline
GPU 1
GPU 2
GPU 3
GPU 4
GPU 5

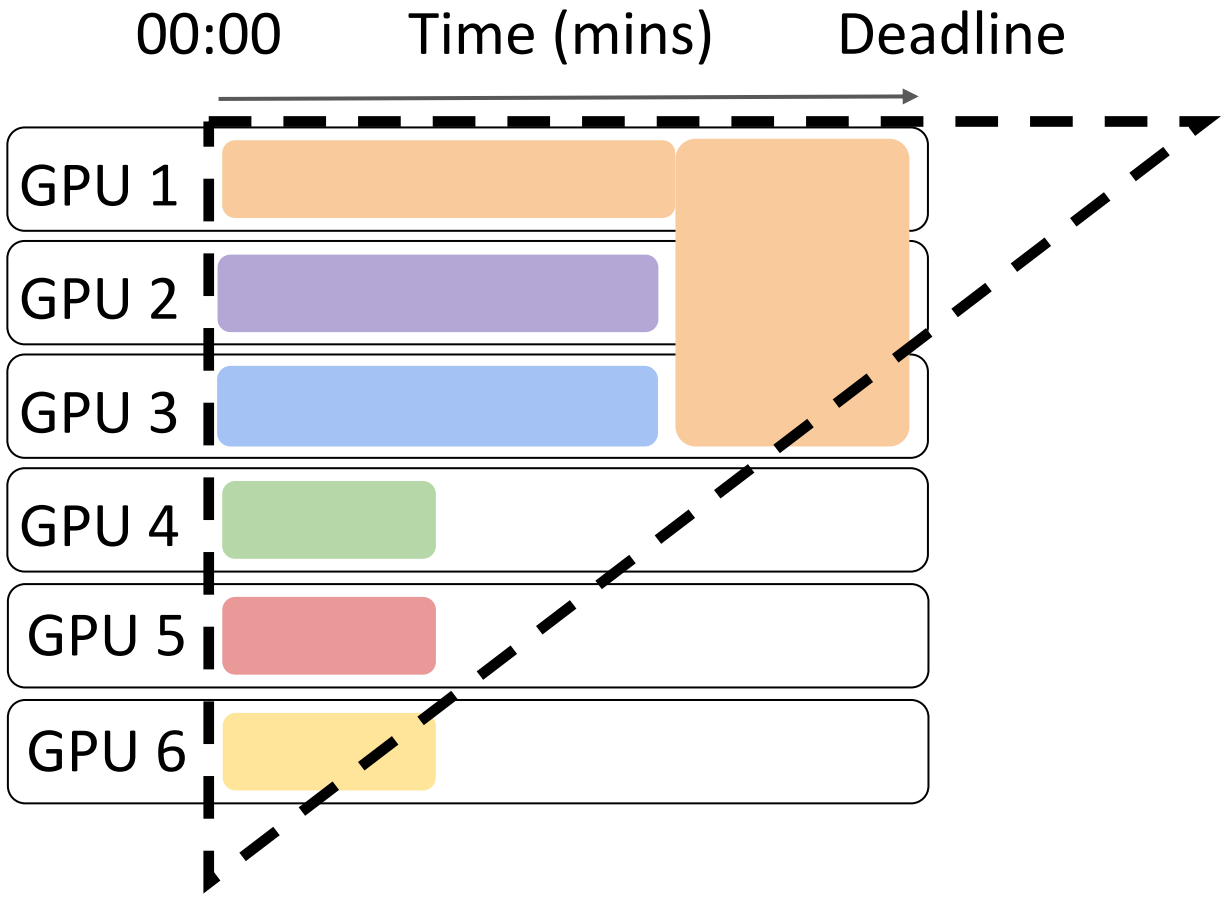

00:00
Time (mins)
Deadline
GPU 1
GPU 2
GPU 3
GPU 4
GPU 5
GPU 6


00:00
Time (mins)
Deadline
GPU 1
GPU 2
GPU 3
GPU 4
GPU 5
GPU 6


00:00
Time (mins)
Deadline
GPU 1
GPU 2
GPU 3
GPU 4
GPU 5
GPU 6

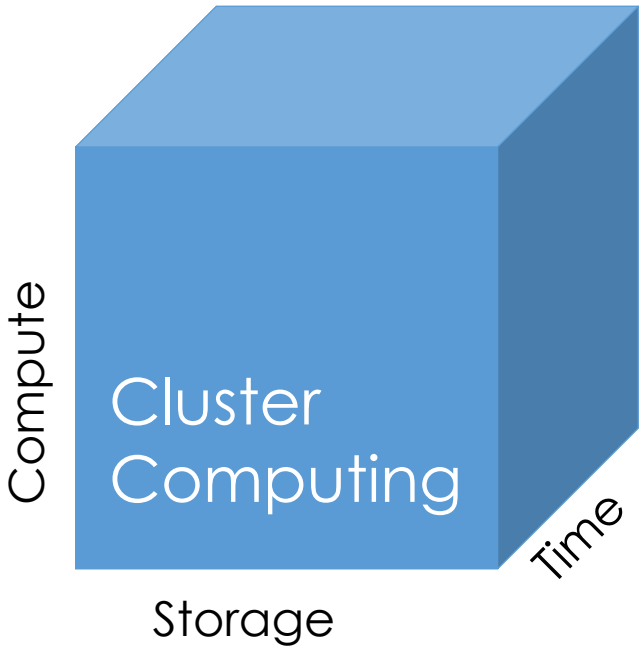
When it comes to machine-time allocation

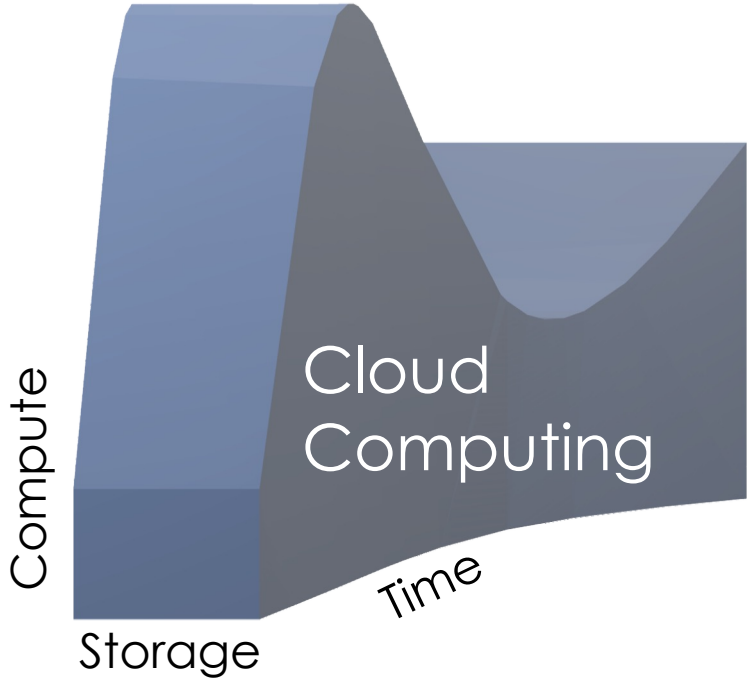




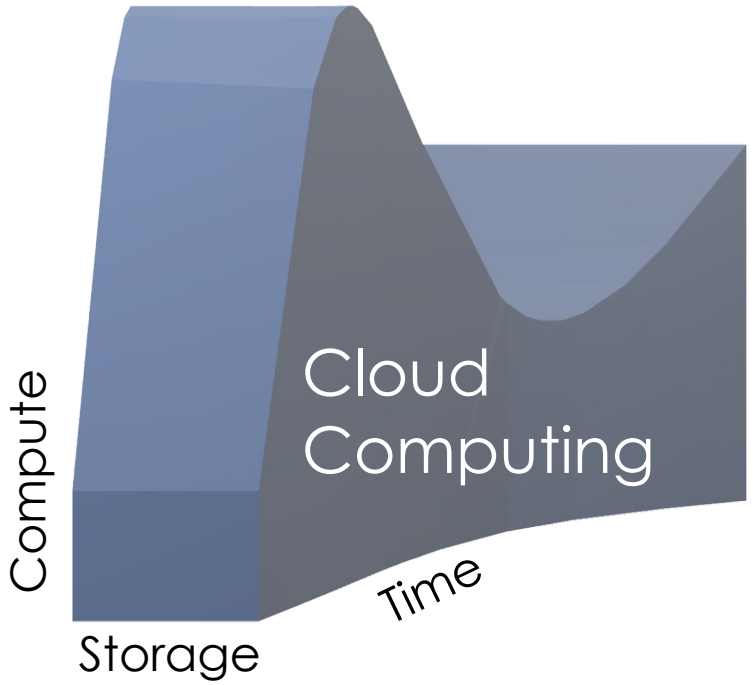
Triangles > Rectangles

Think outside the box





Cloud computing → Infinite resources with a finite budget (volume constraints).



The cloud as a utility

"Serverless"

Serverless Computing: One Step Forward, Two Steps Back

Joseph M. Hellerstein, Jose Faleiro, Joseph E. Gonzalez, Johann Schleier-Smith, Vikram Sreekanti,
Alexey Tumanov and Chenggang Wu
UC Berkeley
{hellerstein,jmfaleiro,jegonzal,jssmith,vikrams,atumanov,cgwu}@berkeley.edu

ABSTRACT

Serverless computing offers the potential to program the cloud in an autoscaling, pay-as-you go manner. In this paper we address critical gaps in first-generation serverless computing, which place its autoscaling potential at odds with dominant trends in modern computing: notably data-centric and distributed computing, but also open source and custom hardware. Put together, these gaps make current serverless offerings a bad fit for cloud innovation and particularly bad for data systems innovation. In addition to pinpointing some of the main shortfalls of current serverless architectures, we raise a set of challenges we believe must be met to unlock the radical potential that the cloud—with its exabytes of storage and millions of cores—should offer to innovative developers.

1 INTRODUCTION

Amazon Web Services recently celebrated its 12th anniversary, marking over a decade of public cloud availability. While the cloud began as a place to timeshare machines, it was clear from the beginning that it presented a radical new computing platform: the biggest assemblage of data capacity and distributed computing power ever available to the general public, managed as a service.

Despite that potential, we have yet to harness cloud resources in radical ways. The cloud today is largely used as an outsourcing platform for standard enterprise data services. For this to change, creative developers need programming frameworks that enable

offers the attractive notion of a
opers simply upload their co
their behalf as needed at any
themselves with provisioning
only for the compute resource

The notion of serverless c
optimists to project any numb
on what it might mean. Our g
terminology. Concretely, each
launched serverless computin
significant marketing budget p
the field based on the serverle
are actually offering today and
when viewed in light of the cl

1.1 "Serverless" goes

To begin, we provide a quick i
(FaaS), the commonly used and
of serverless offerings from t
AWS was the first public cloud
our discussion on the AWS F
from Azure and GCP differ in

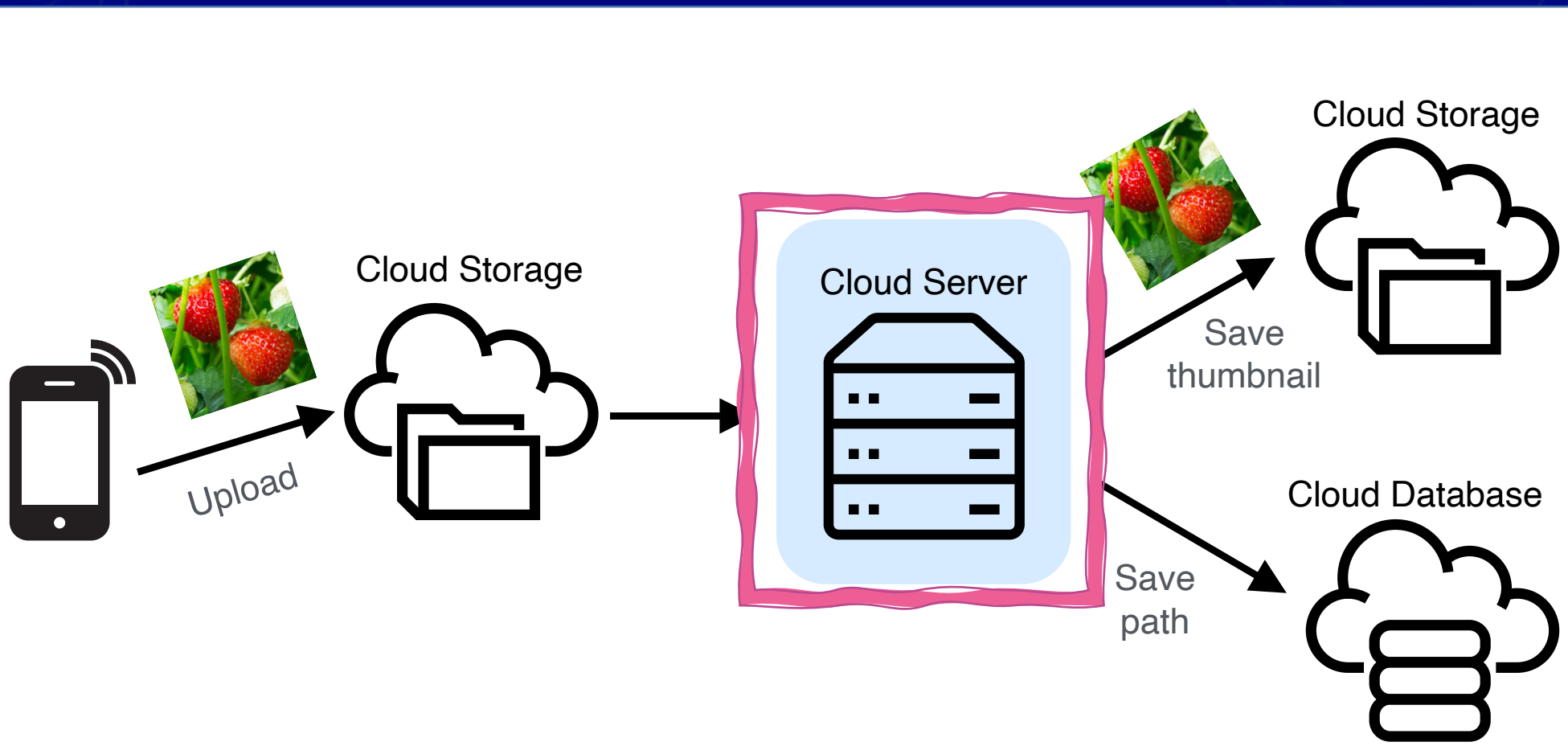
The idea behind FaaS is simp
textbook. Traditional program
which are mappings from inp

A Berkeley View on Serverless Computing -- Cloud Programming Simplified

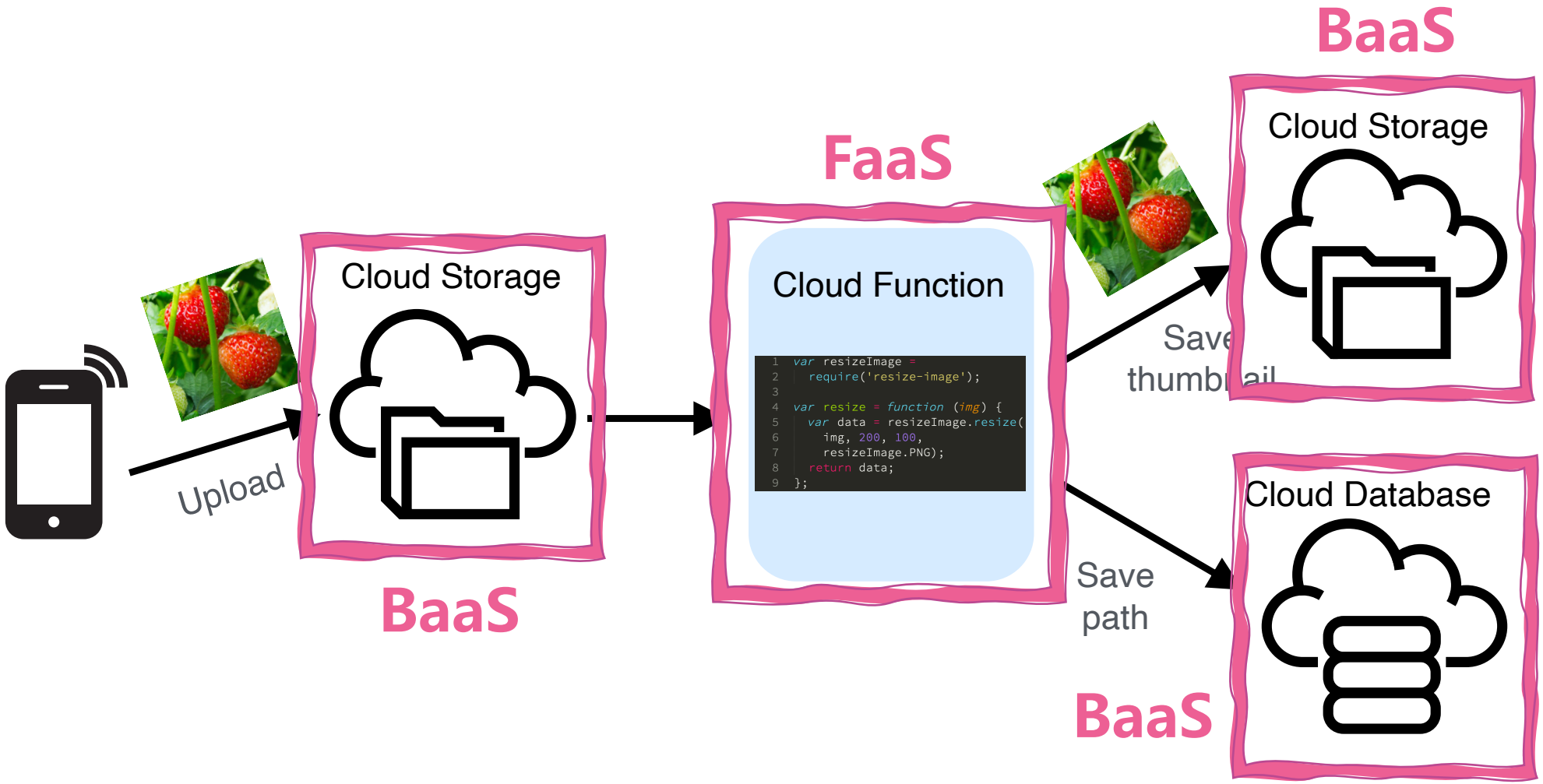
Eric Jonas, Johann Schleier-Smith, Vikram Sreekanti, Chia-che Tsai, Anurag Khandelwal,
Qifan Pu, Vaishaal Shankar, Joao Carreira, Karl Krauth, Neeraja Yadwadkar,
Joey Gonzalez, Raluca Ada Popa, Ion Stoica and David Patterson

Abstract: Serverless cloud computing handles virtually all the system administration operations needed to make it easier for programmers to use the cloud. This paper gives a quick history of cloud computing, explains the motivation for serverless computing, describes applications that stretch the current limits of serverless, and then lists obstacles and research opportunities required for serverless computing to fulfill its full potential. Just as the Berkeley View of Cloud Computing paper identified challenges for the cloud in 2009 and predicted they would be addressed and that cloud use would accelerate, we predict these issues are solvable and that serverless computing will grow to dominate the future of cloud computing.

Canonical example



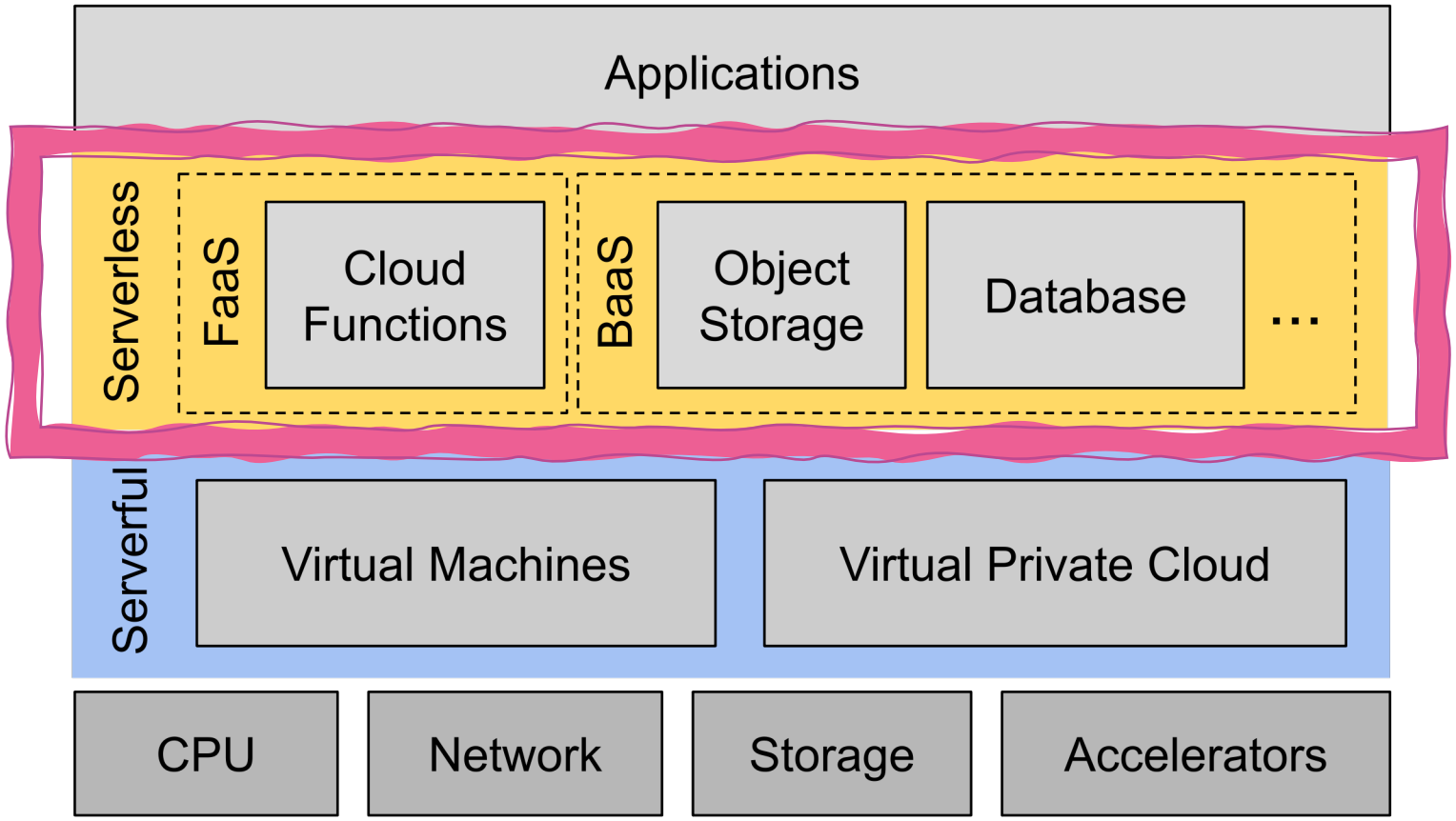
Canonical example



AWS Lambda

- Introduced Function as a Service (FaaS)
- Autoscaling done right
  - Highly elastic - adapts quickly
  - Scales down to zero
  - Fine-grained 100 ms billing increment
  - Cloud provider shares risk and responsibility for utilization
- Strong isolation allowing multi-tenant multiplexing
- Benefits from scale of Amazon's platform & ecosystem of APIs

A Layer to Simplify Using the Cloud



Three essential qualities of serverless computing

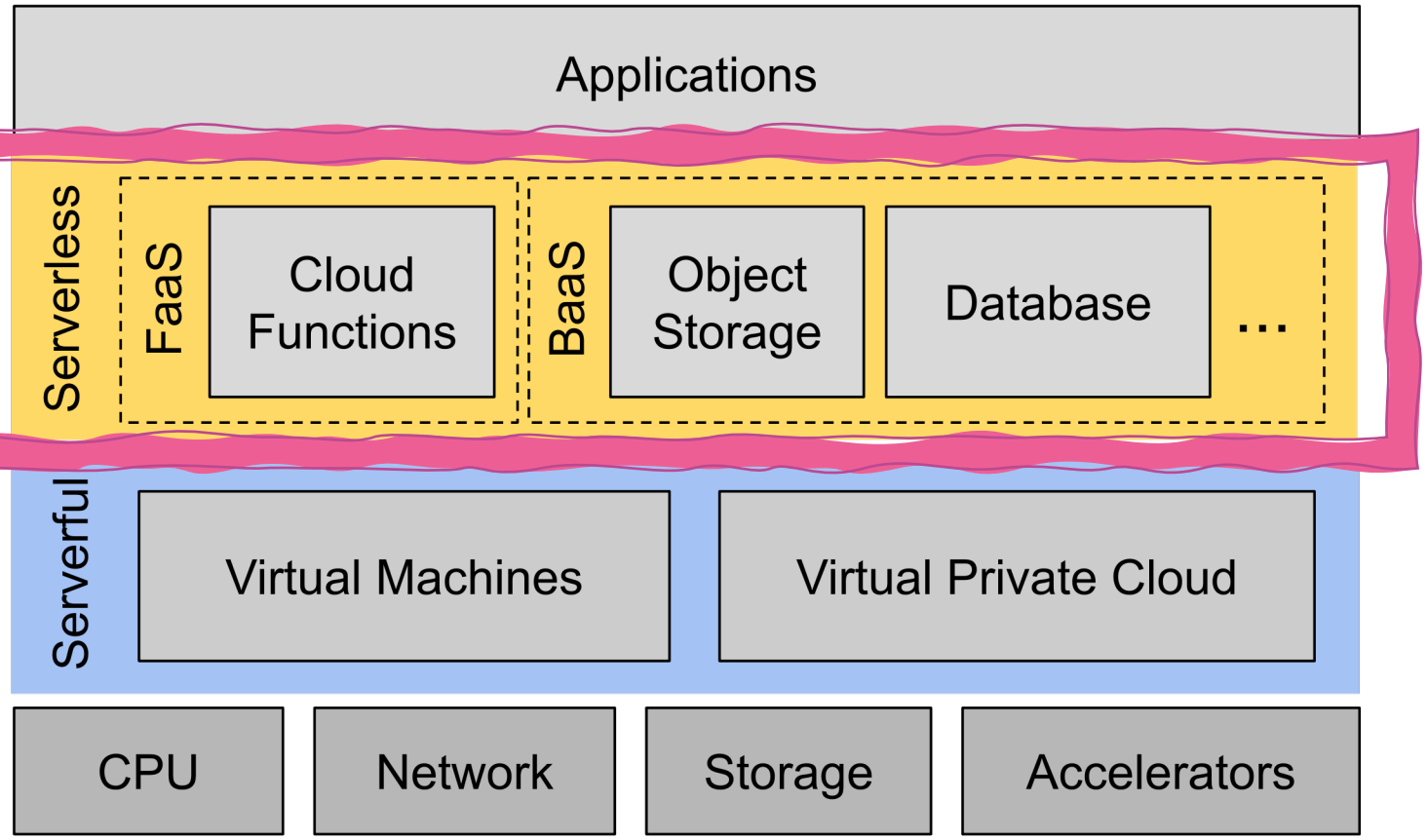
- **Hides servers** and the **complexity** of programming and operating them
- Offers a **pay-per-use cost** model with no charge for idle resources
- Has **excellent autoscaling** so resources match demand closely

Airport Analogy

When you arrive at the destination airport and need to get to your hotel you could:

1. **Buy a car** and drive [Legacy on premise systems]
   - Long term investment and you are responsible for everything
2. **Rent a car** and drive [Serverfull]
   - You are still responsible for fuel, parking, insurance, …
3. **Take an Uber**. [Serverless]
   - You are paying only for the transportation you need

Where is the cloud headed?



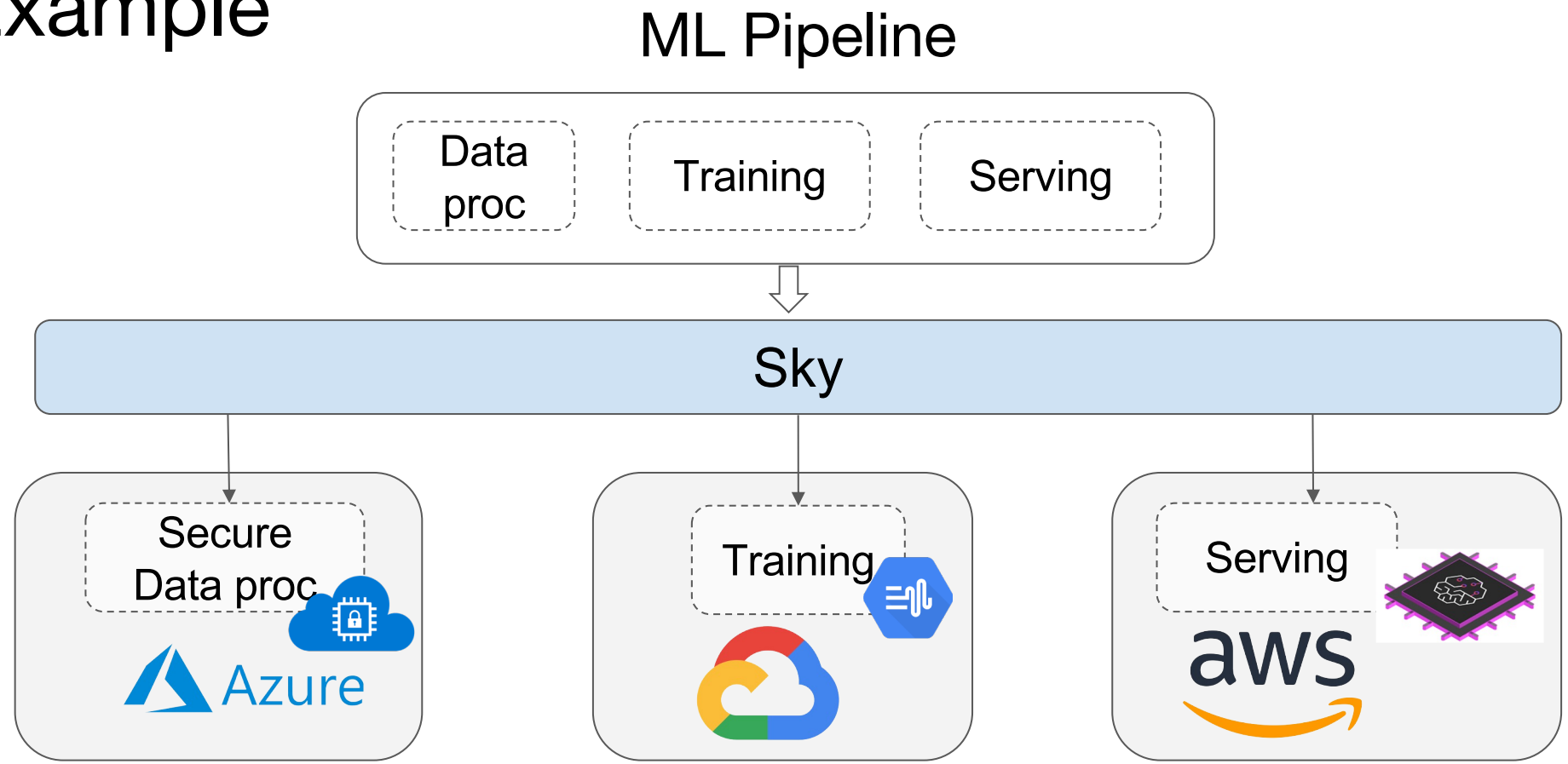
With each phase of the cloud (and computing), we **raised the abstraction**.

This continuous process of simplification enables **concurrent innovation on both sides of the boundary**.

It also **improves portability** and **transforms the market**.


Sky Computing

Sky Example



- Use Azure Confidential Computing for secure data processing
- Use Google Cloud for training on TPUs (fastest and cheapest)
- Use AWS for serving on Inferentia (cheapest)

Conjectures for the Future

- We will continue to race towards utility-oriented computing
  - Pay for consumption and not capacity
- Higher levels of abstraction will
  - **Reduce operational complexity** (burden shifts to the cloud + ML)
  - Drive more **rapid innovation in cloud hardware**
  - Enable applications to more easily **span multiple clouds**
- **Sky computing** is the inevitable future of computing

Readings This Week

Pollux: Co-adaptive Cluster Scheduling for Goodput-Optimized Deep Learning

- Published in OSDI'21 – **Best Paper Award**
- Why we chose it?
  - Good example of work exploring **scheduling of for ML**
  - Addresses **ML and Systems concerns:** throughput, improvements in accuracy, fairness?
- Things to think about:
  - Implications for elasticity?
  - What about hyperparameter search?

The Sky Above the Clouds [Unpublished]

- Draft of the **vision paper** describing **Sky research agenda**
  - *Do Not Distribute*
  - Feedback will help the paper (be critical!)
- Makes a case for both the **inevitability** and **need for research** in "Sky Computing"
- Things to think about:
  - Presentation of premise [what is proposed?]
  - Role of data
  - Role of research
  - ML Systems Research Case?

FrugalML: How to Use ML Prediction APIs More Accurately and Cheaply

- Published in NuerIPS'20
- Example of a "Sky Computing" ML research direction
  - **Combining competing prediction services** to improve accuracy and reduce costs.
  - Potentially exciting new research direction!
- Things to think about:
  - Latency
  - Out-of-domain performance
  - Uncertainty calibration and model biases
  - Mathematical presentations