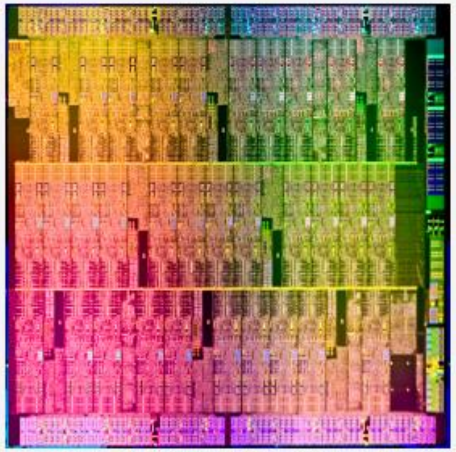
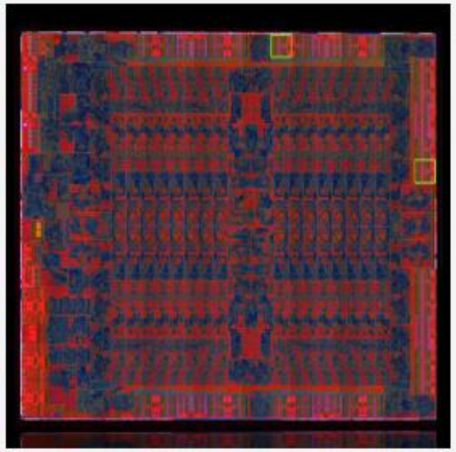
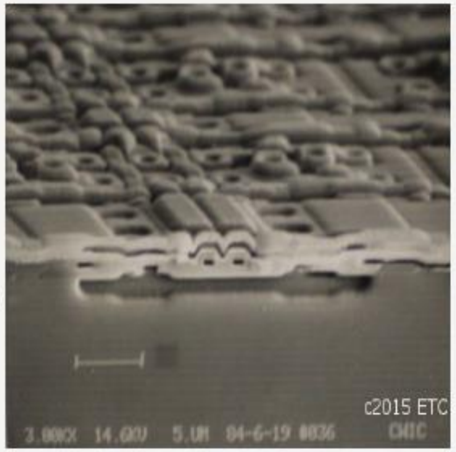
18-344: Computer Systems and the Hardware-Software Interface

Fall 2023







Course Description

This course covers the design and implementation of computer systems from the perspective of the hardware software interface. The purpose of this course is for students to understand the relationship between the operating system, software, and computer architecture. Students that complete the course will have learned operating system fundamentals, computer architecture fundamentals, compilation to hardware abstractions, and how software actually executes from the perspective of the hardware software/boundary. The course will focus especially on understanding the relationships between software and hardware, and how those relationships influence the design of a computer system's software and hardware. The course will convey these topics through a series of practical, implementation-oriented lab assignments.

Credit: Brandon Lucia

Feedback discussion

- Course content & lectures are interesting + good pace
- Course structure
- TA support
- Regular feedback
- HW + labs – very practical + relevant to course content
- Lab writeups have become better over time + addition of rubrics
    - Design flexibility
- Participation encouragement
- Availability of slides

Feedback discussion

- Posting slides more regularly
- More specific report writing guidelines – report guidelines posted on webpage?
- Made aware of how much time the labs take? Some labs take time
- Earlier labs due sooner, to get more time for the later labs?
  - Challenge: Can release a lab only after covering relevant content in the lecture
- Better explanation of tools used in the labs – step-by-step process for how to use the tools + expected results for each benchmark
  - Need more detail
- Late day policy not explained – please check lecture 1
- Task breakups for all labs – based on prior feedback
- Some more practice problems – in what form? – example practice probs in recitation
- Lab checkpoints? – rather hard to implement – send to Tas
- More helpful recitations – specifics? – recitation slides posted
- Page limits for lab reports – how is this helpful since it's going to make you reduce # images/size?

Feedback discussion

- Quicker grading of HWs – benefit?
- Made aware of how much time the labs take? Some labs take time
- Partial credit for some HW probs – I think this is already awarded? Example: Q1.2
  - Please bring to TAs' attention if you don't receive partial credit
- Piazza Qs are answered slower than Slack
- Getting stuck on Qs asked out to the class
  - Don't worry about not having an answer; it's just to get you thinking
- More descriptive lab handouts – specifics?
- Classroom is cold
- More OH – happy to do on-demand ones
- Lab partner selection process – explain?
- Each HW is high stakes (5% of grade)? – each HW is only 2% of the grade – update website with HW due dates
- Auto-grader – unfortunately, can't implement this time - usually takes a few months to get this right

Mid-term grades rubric

- 90%+ is A
- 85-90% is –
- 80-85% is B+
- 70-80% is B
- 60-70% is B-
- Any other forms for feedback you're expecting?

Agenda

INTRODUCTION TO DATA CENTER COMPUTING

TREND 1: DECLINE IN HARDWARE PERFORMANCE SCALING

TREND 2: GROWING DATA MOVEMENT OVERHEADS

TRENDS 3, 4, 5: RAPID GROWTH IN DATA, USERS, FUNCTIONALITY

TREND 6: DISCONNECT BETWEEN CPU & I/O

TREND 7: A COMPLEX DESIGN SPACE

Have you used the cloud before?

Carnegie Mellon University

Web services are everywhere

Are web services free?

I can use Twitter (X?) for free!

#thebestthingsinlifearefree

Users use most services for free *

* Fine print: Not really free

What do web services require?

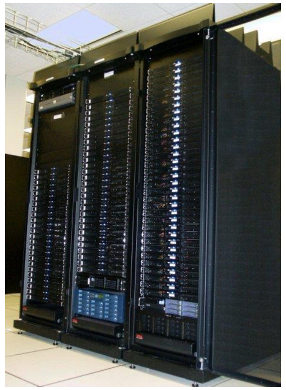
Web service computation requirements

- How many computers do web services need?
  - Computation requirement: Poll
  - Facebook: Hundreds of thousands of servers
  - Intel: Hundreds of thousands of servers in 97 data centers
    - AMD and ARM are quickly catching up; why so?
  - Google: >1M servers & is planning for 10M
    - Why might Google need more servers than Facebook?
- *What does this scale of operation mean for you as a systems researcher?*

Idea behind “scaling up” vs. “scaling out”

PC

Server



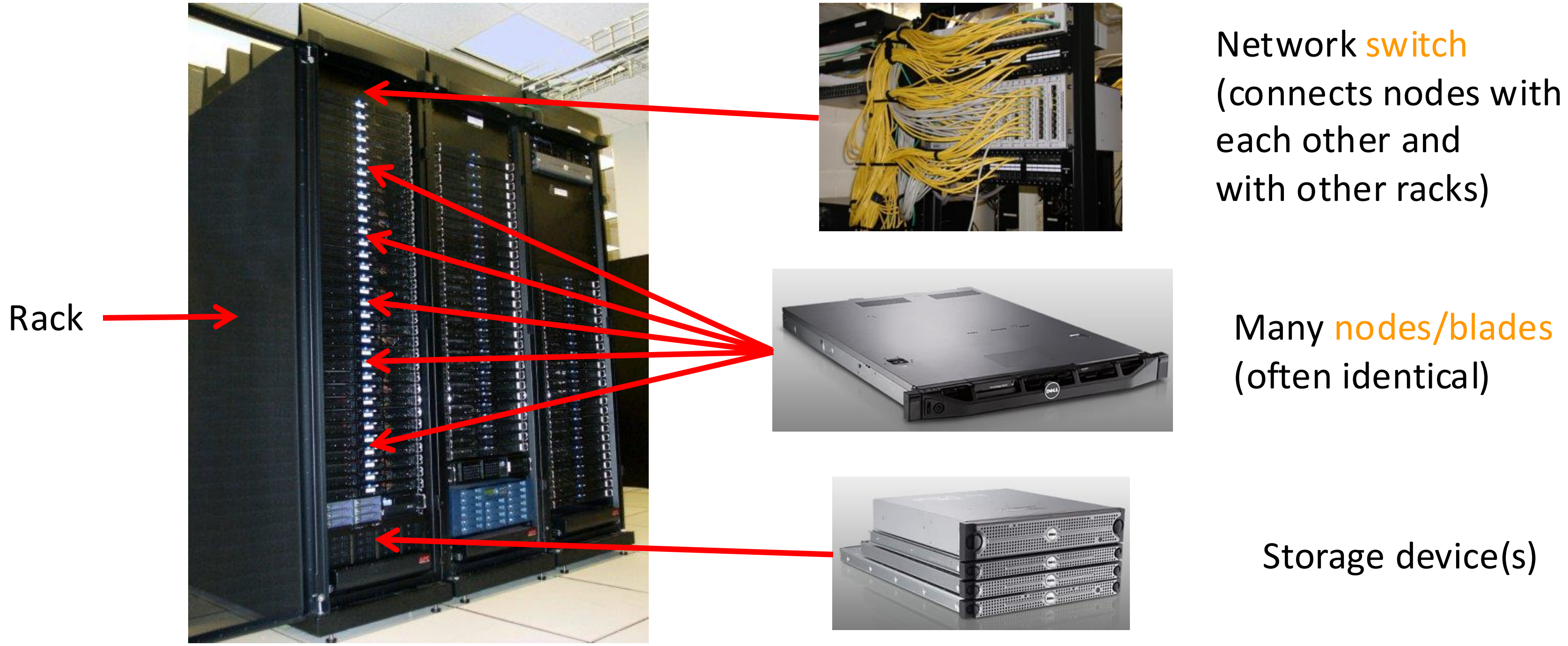
Cluster

- What if one computer is not enough?
  - Buy a bigger (server-class) computer

- What if the biggest computer is not enough?
  - Buy many computers

Carnegie Mellon University

Clusters



- Characteristics of a cluster:
  - Many similar machines, close interconnection (same room?)
  - Often special, standardized hardware (racks, blades)
  - Usually owned & used by a single organization

Power & cooling for clusters

- Clusters need lots of power
  - Example: 140 Watts per server
  - 32 server rack: 4.5kW (needs special power supply!)
  - Most power -> heat

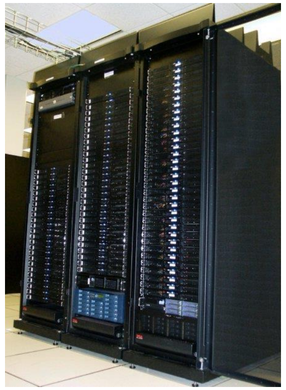
- Large clusters need massive cooling
  - 4.5kW is ~3 space heaters
  - And that's just one rack!

Carnegie Mellon University

More scaling up

PC

Server



Cluster

Data center

- Have you seen a cluster?
- Which is the oldest cluster?
- What if your cluster is too big (hot, power hungry) to fit into your office building?
    - Build a separate building for the cluster
    - Building can have lots of cooling and power
    - Result: Data center

What does a data center look like?

Google campus in The Dalles, Oregon

- A warehouse-sized computer
  - A single data center can easily contain 10K racks with 100 cores in each rack (1M cores total)

Even more scaling up!

PC

Server

Cluster

Data center

Network of data centers

- What if even a data center is not big enough?
  - Build additional data centers
  - Where? How many?

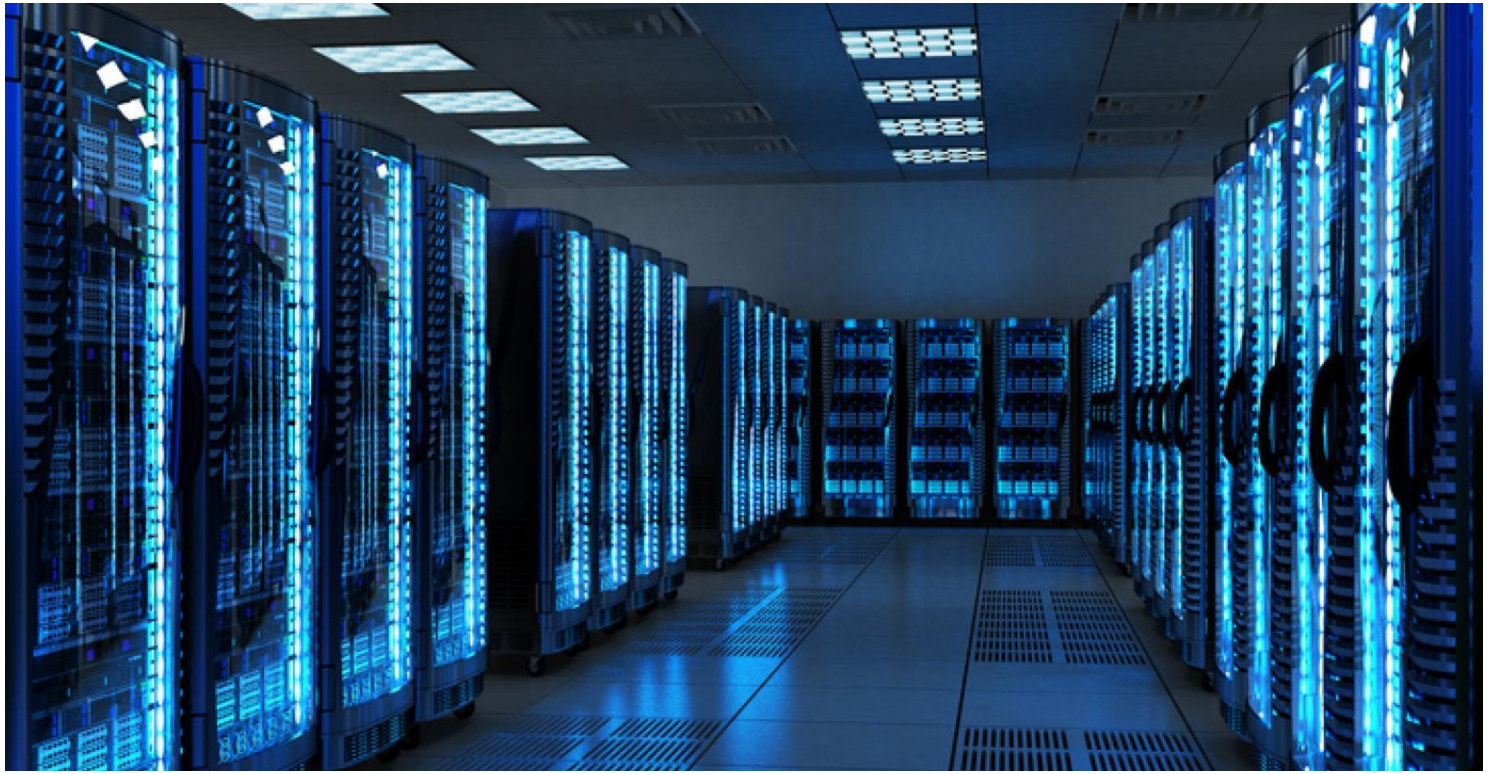
Data centers here, there, everywhere!

- Data centers are often globally distributed
  - Example above: Google data center locations
- Why?
  - Need to be close to users (physics!)
  - Cheaper resources
  - Protection against failures

Web services are everywhere

amazon

Uber

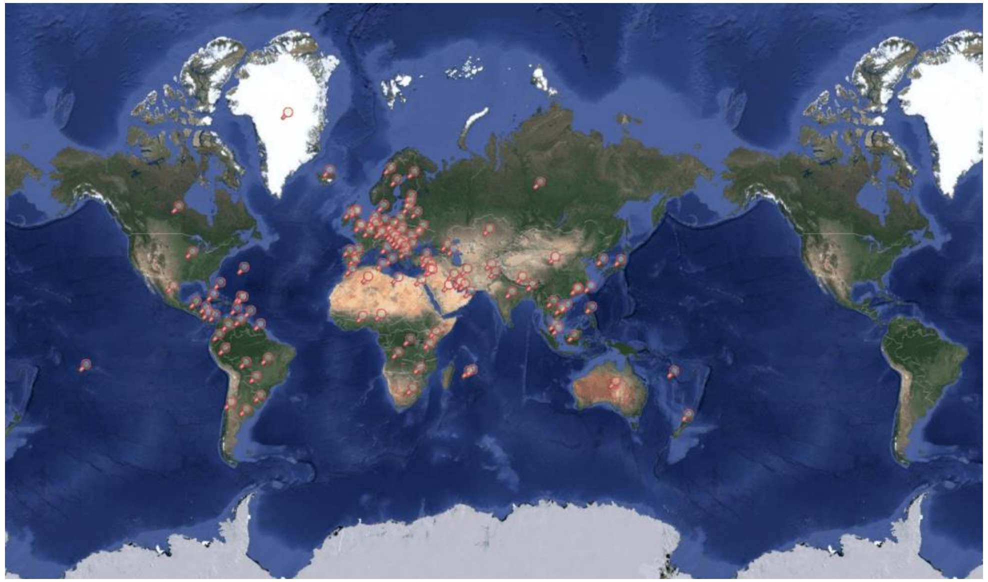


Web services run across hundreds of thousands of servers, at hyperscale

What does hyperscale computing require today?

Carnegie Mellon University

How much do data centers cost?



$100s of millions to build each data center

[cbinsights.com/research/future-of-data-centers]

X 142

Low cost:

Facilitate new services

How much power do data centers require?

3% of net energy [Masanet '18]

Low energy:
Sustainability (go green!)

How do data centers impact sustainability?

How fast do web services have to be?

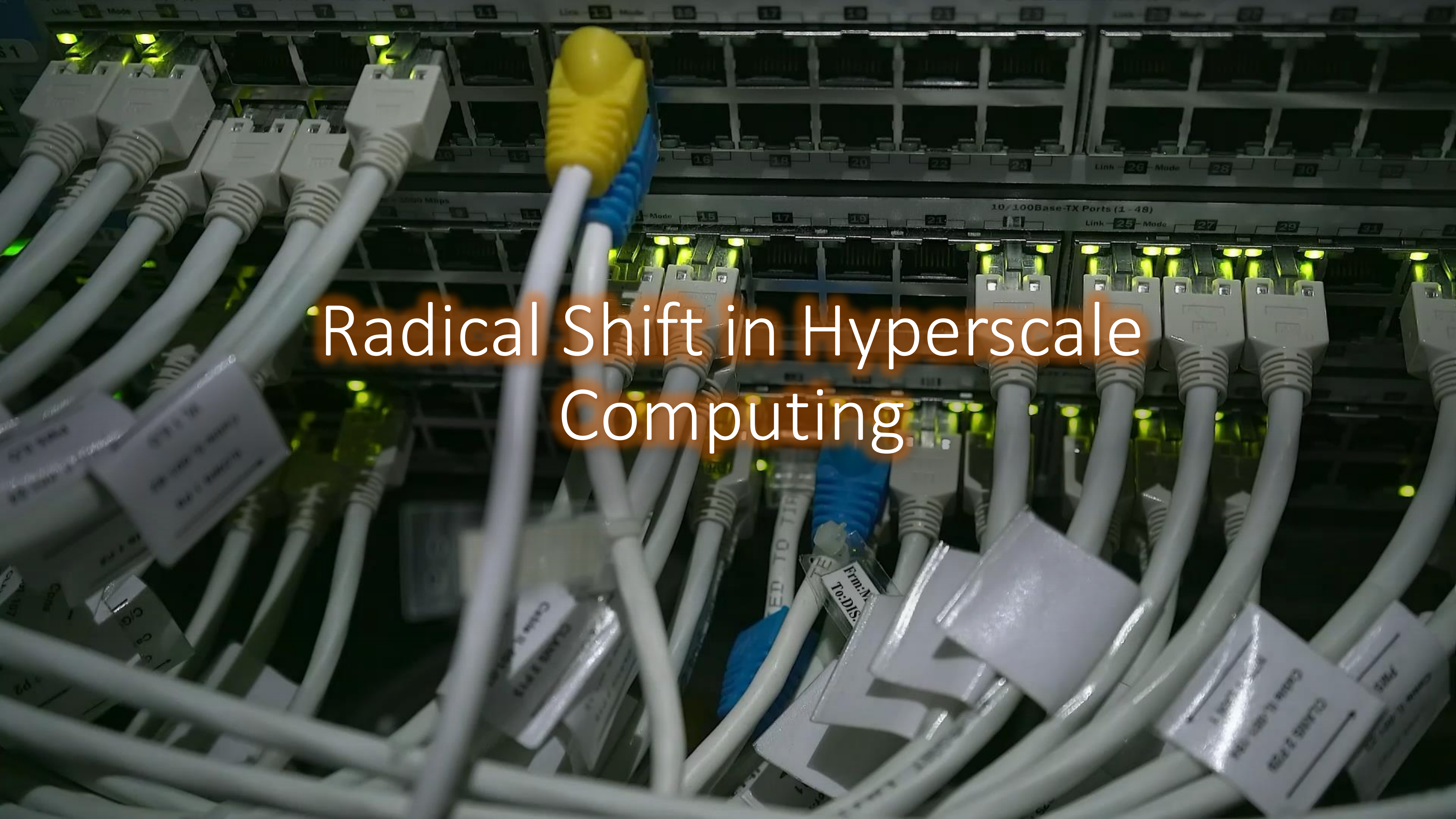
~Instantaneous

High Performance:
Retain users

Can we keep sustaining data centers this way?

Carnegie Mellon University

Radical Shift in Hyperscale Computing

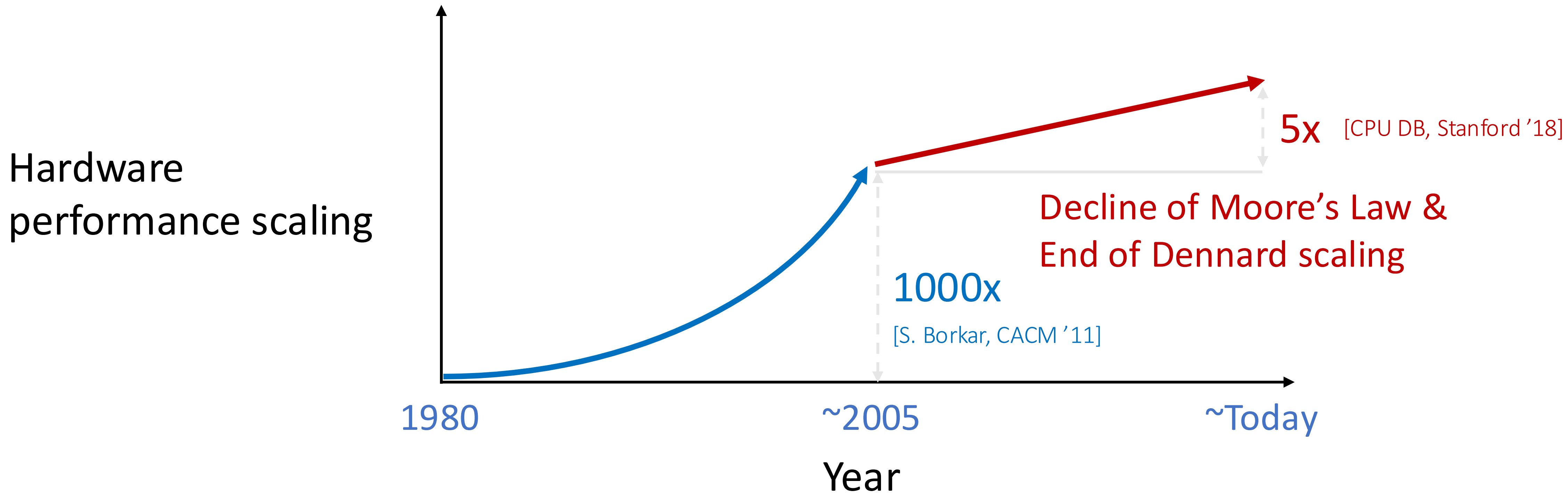


Agenda

- INTRODUCTION TO DATA CENTER COMPUTING
- **TREND 1: DECLINE IN HARDWARE PERFORMANCE SCALING**
- TREND 2: GROWING DATA MOVEMENT OVERHEADS
- TRENDS 3, 4, 5: RAPID GROWTH IN DATA, USERS, FUNCTIONALITY
- TREND 6: DISCONNECT BETWEEN CPU & I/O
- TREND 7: A COMPLEX DESIGN SPACE

Radical Shift in Hyperscale Computing

Trend 1: Decline in HW performance scaling



Challenge: Traditional servers offer diminishing performance returns

Carnegie Mellon University

Trend 1: Decline in HW performance scaling

How has systems research adapted?

HARE: Hardware Accelerator for

Vaibhav Gogte*, Aasheesh K

and Thomas F. Wenisch*

Abstract—Rapidly processing text data is critical for many

Conventional software solutions for regexp processing are

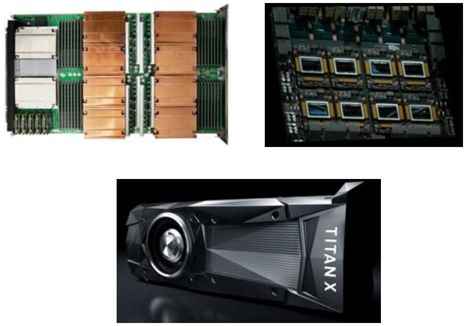
Abstract—Feature extraction is an essential part in applications that require computer vision to recognize objects in an image

ment in terms of noise immunity and invariance to intensity change and rotation.

Abstract—Interactive AI-powered services require low-latency evaluation of deep neural network (DNN) models—aka "real-time AI". The growing demand for computationally expensive,

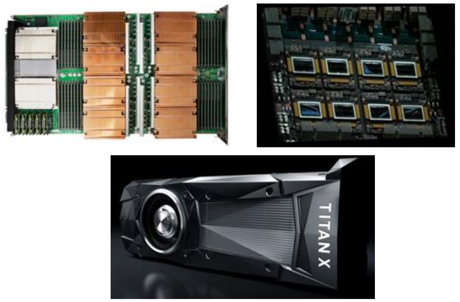
be much more latency sensitive. DNNs are increasingly used

Abstract—Deep rapidly becoming a major compon



Custom hardware (e.g., GPU)

However: Diverse accelerators will break the bank

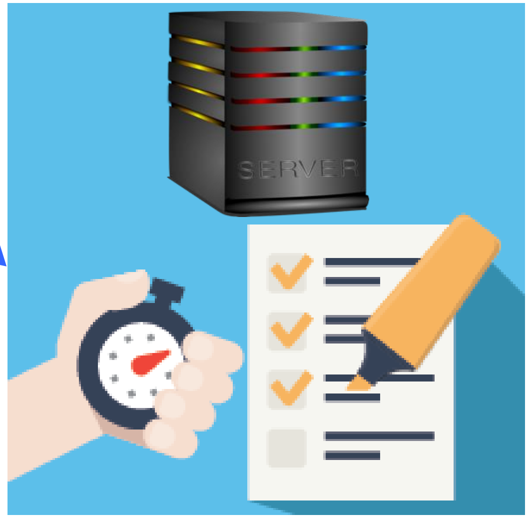


Procurement @scale

Customized platforms -> expensive

Hardware homogeneity

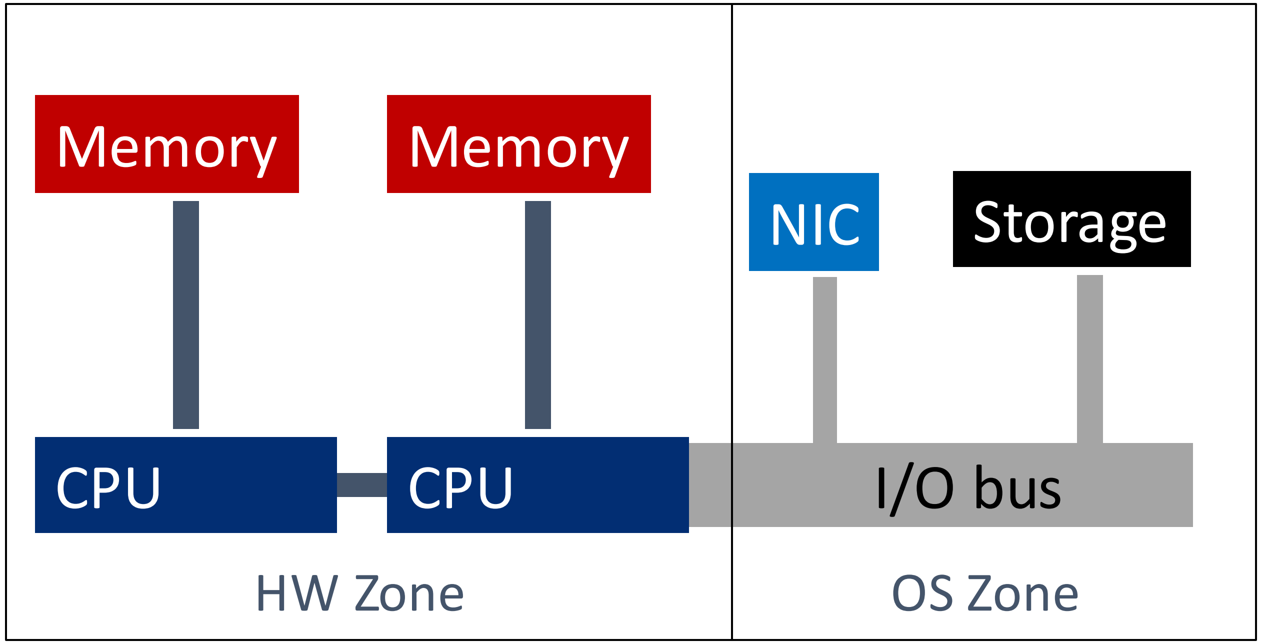
Avoids testing overhead



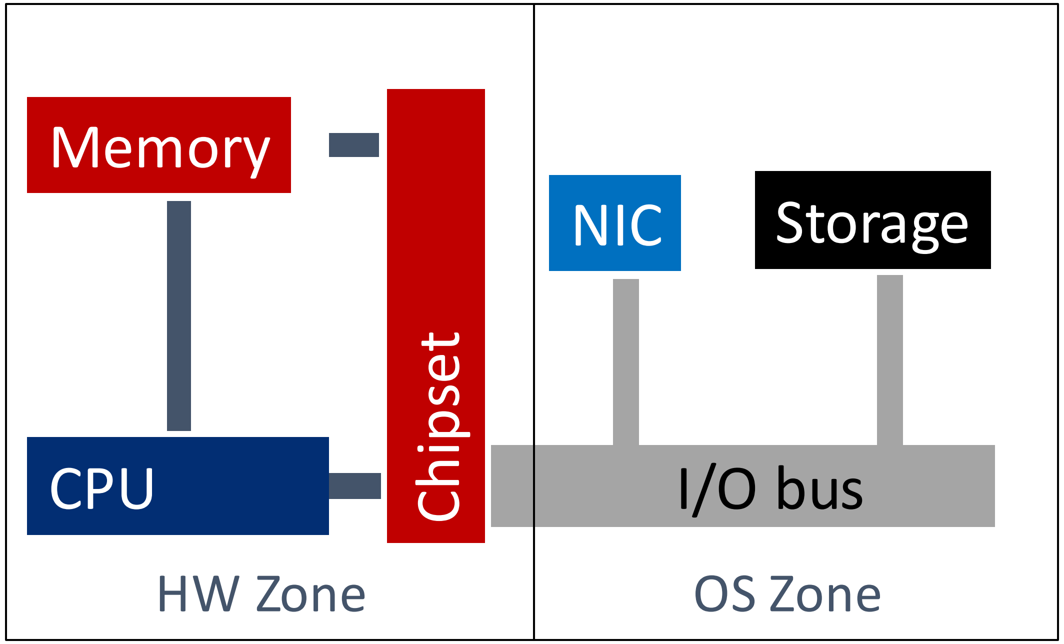
Must think carefully about how we must architect data center hardware

Carnegie Mellon University

What do today's servers look like?



Today's server blade



Desktop of the 1980's

Today's server blades resemble the desktop PCs of the 1980s!

Carnegie Mellon University

Agenda

INTRODUCTION TO DATA CENTER COMPUTING

TREND 1: DECLINE IN HARDWARE PERFORMANCE SCALING

TREND 2: GROWING DATA MOVEMENT OVERHEADS

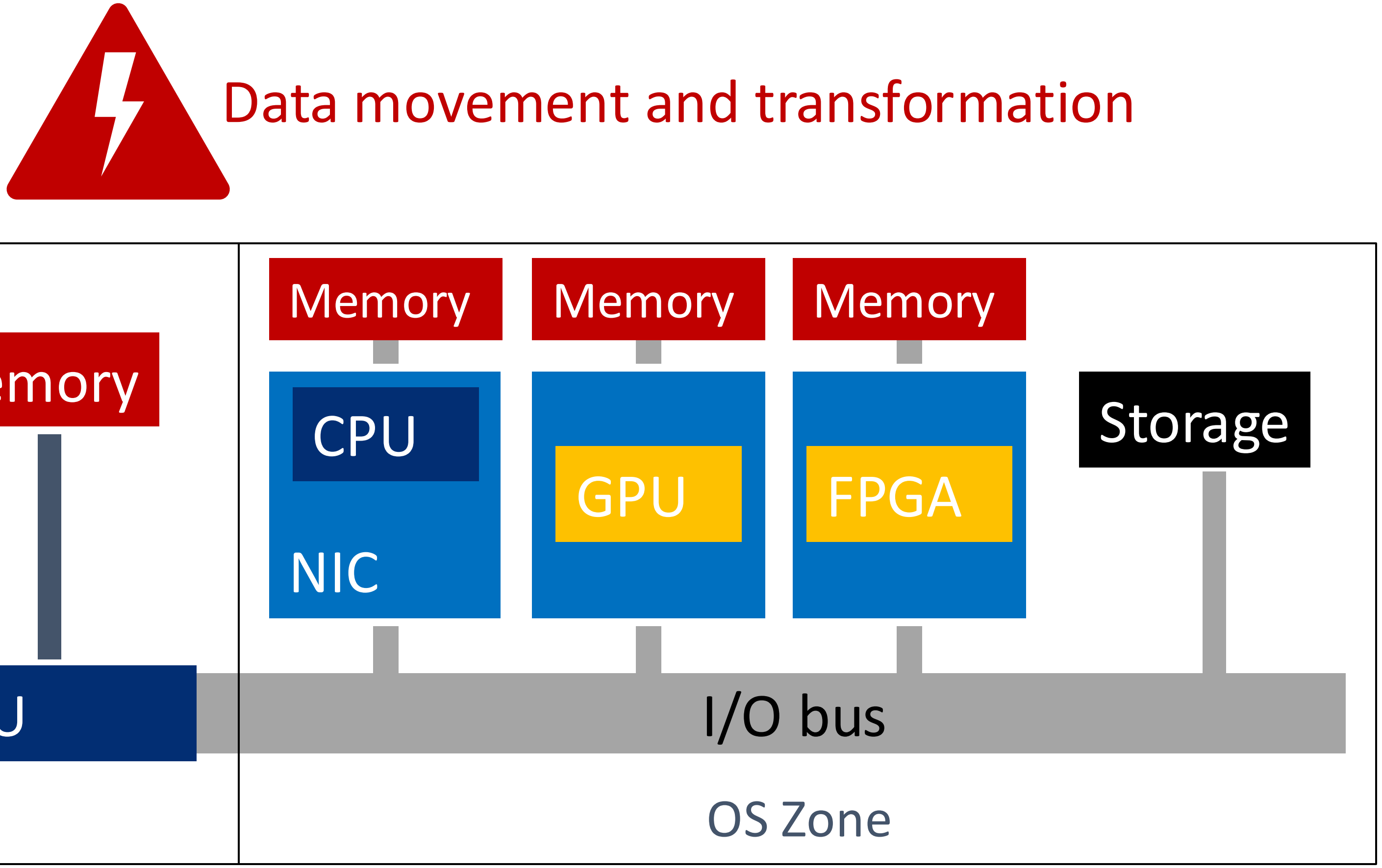
TRENDS 3, 4, 5: RAPID GROWTH IN DATA, USERS, FUNCTIONALITY

TREND 6: DISCONNECT BETWEEN CPU & I/O

TREND 7: A COMPLEX DESIGN SPACE

Radical Shift in Hyperscale Computing

Trend 2: Growing data movement overheads



Today's server blade

Today's server blades resemble the desktop PCs of the 1980s!

Trend 2: Growing data movement overheads

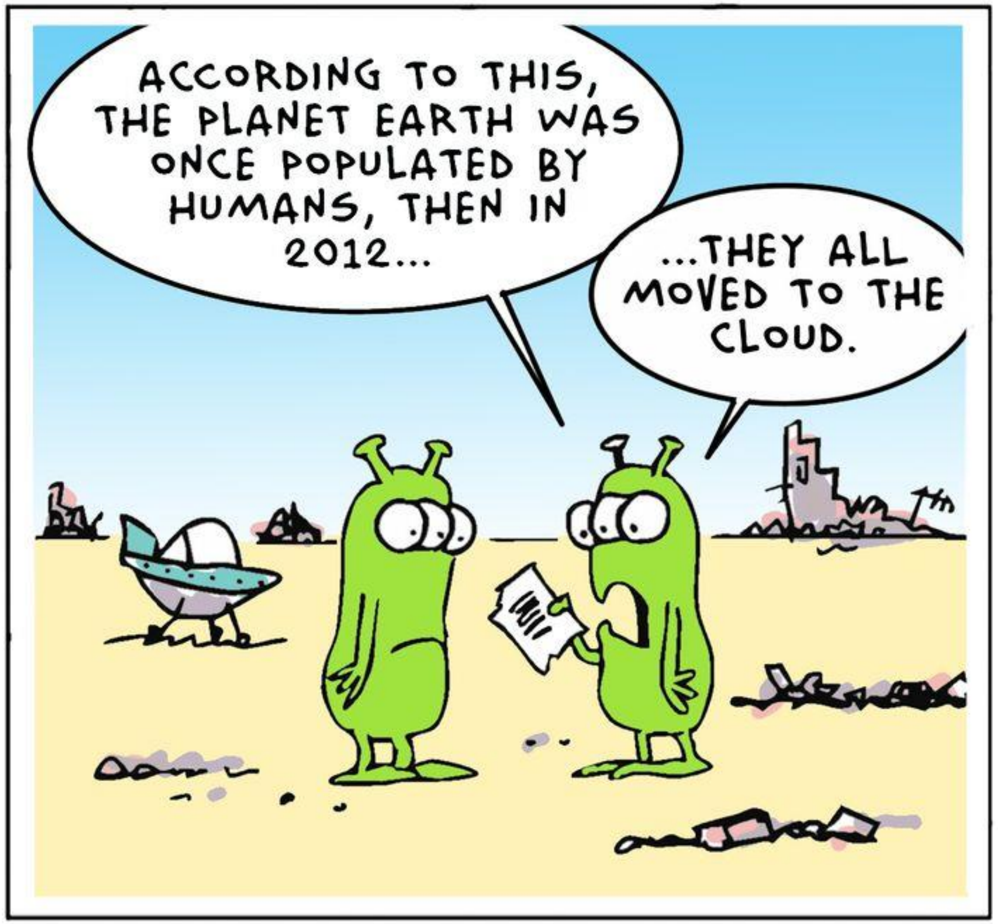
How has systems research evolved?

Transitioning from compute-centric to data-centric architectures

Agenda

- INTRODUCTION TO DATA CENTER COMPUTING
- TREND 1: DECLINE IN HARDWARE PERFORMANCE SCALING
- TREND 2: GROWING DATA MOVEMENT OVERHEADS
- **TRENDS 3, 4, 5: RAPID GROWTH IN DATA, USERS, FUNCTIONALITY**
- TREND 6: DISCONNECT BETWEEN CPU & I/O
- TREND 7: A COMPLEX DESIGN SPACE

Trend 3: A growing user base…



Radical Shift in Hyperscale Computing

Trend 4: Exponential increase in data

Rendering:
>1 petabyte storage

German climate comp. center
60 petabytes of climate data

90 petabytes->
user data

CERN:
>200 petabytes

~10 exabytes

NSA Utah data center
5 zettabyte!!

How much is a zettabyte?

- 1,000,000,000,000,000,000,000 bytes
- Stack of 1TB hard disks that is 25,400 km high

Trend 5: Rapid increase in service functionality

Self-driving cars

Virtual Reality

Conversational AI

…

Carnegie Mellon University

Putting these trends together…

Unprecedented growth
in data, users, & service variety

Challenge: Must support growing data, user base, service functionality

Trends 3, 4, 5: Unprecedented growth in data, users, functionality

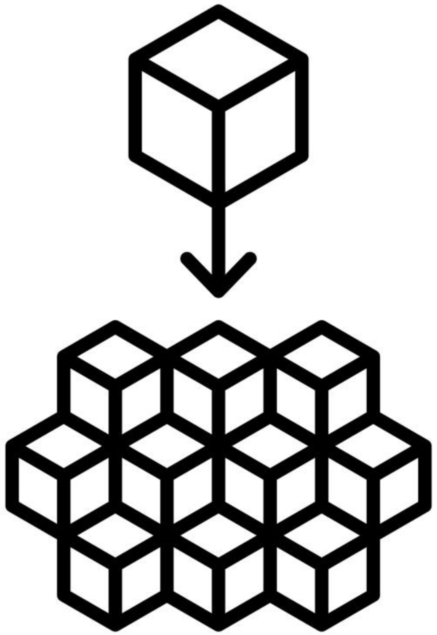
How has the industry adapted?

Data centers here, there, everywhere!

Trends 3, 4, 5: Unprecedented growth in data, users, functionality

How has systems research adapted?

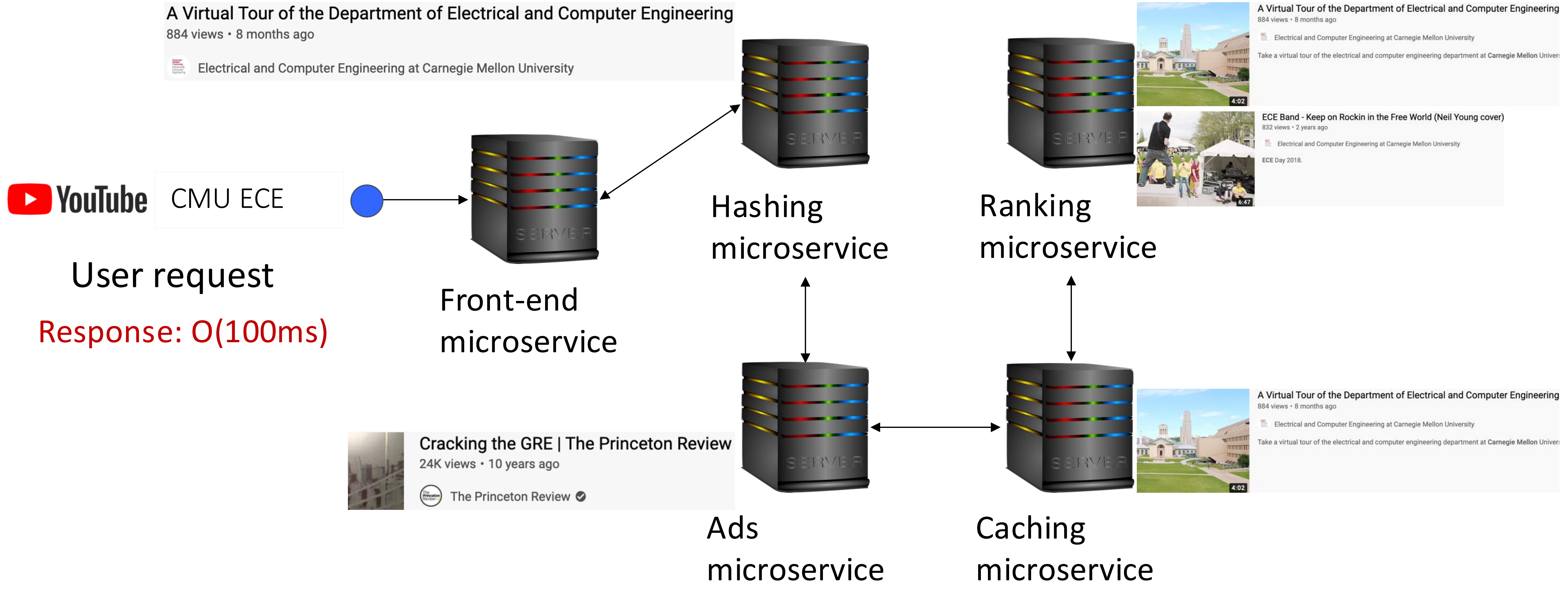
Web service application

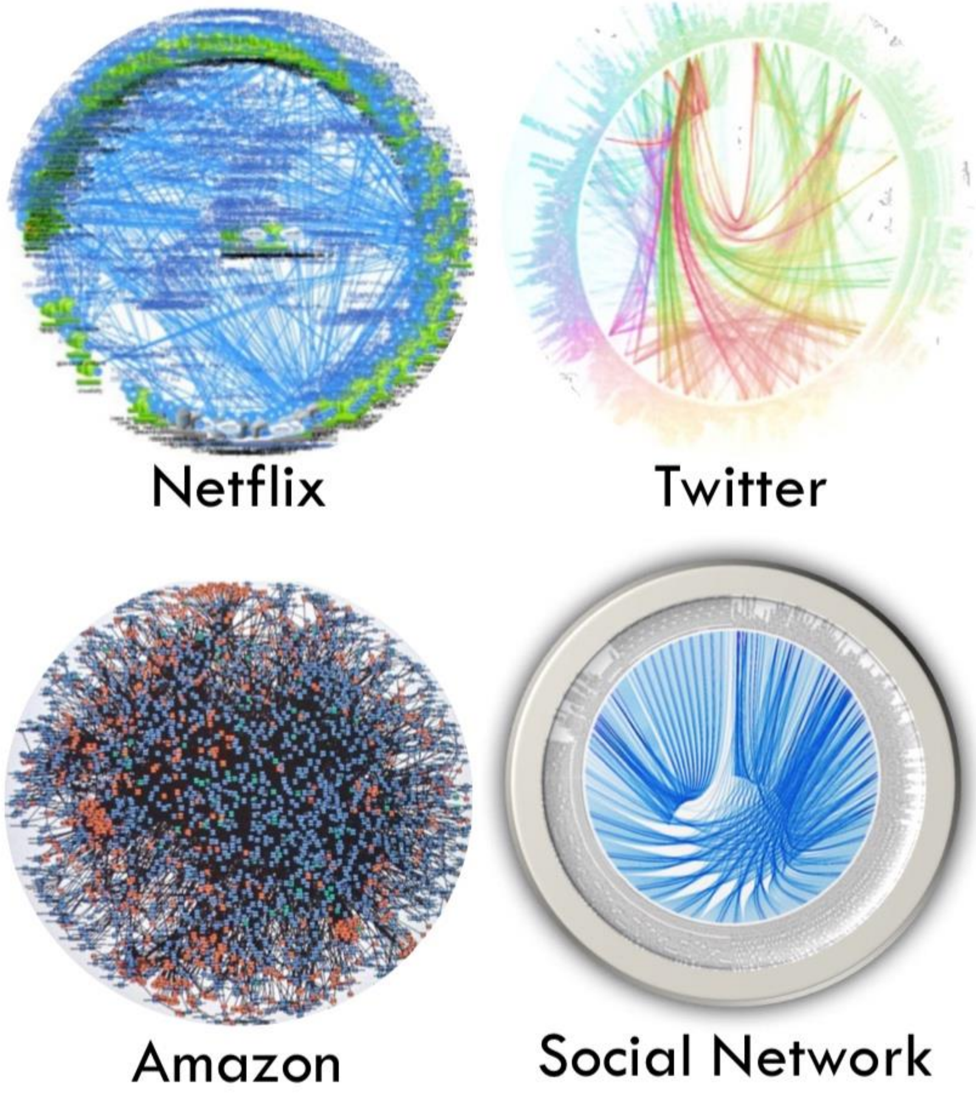


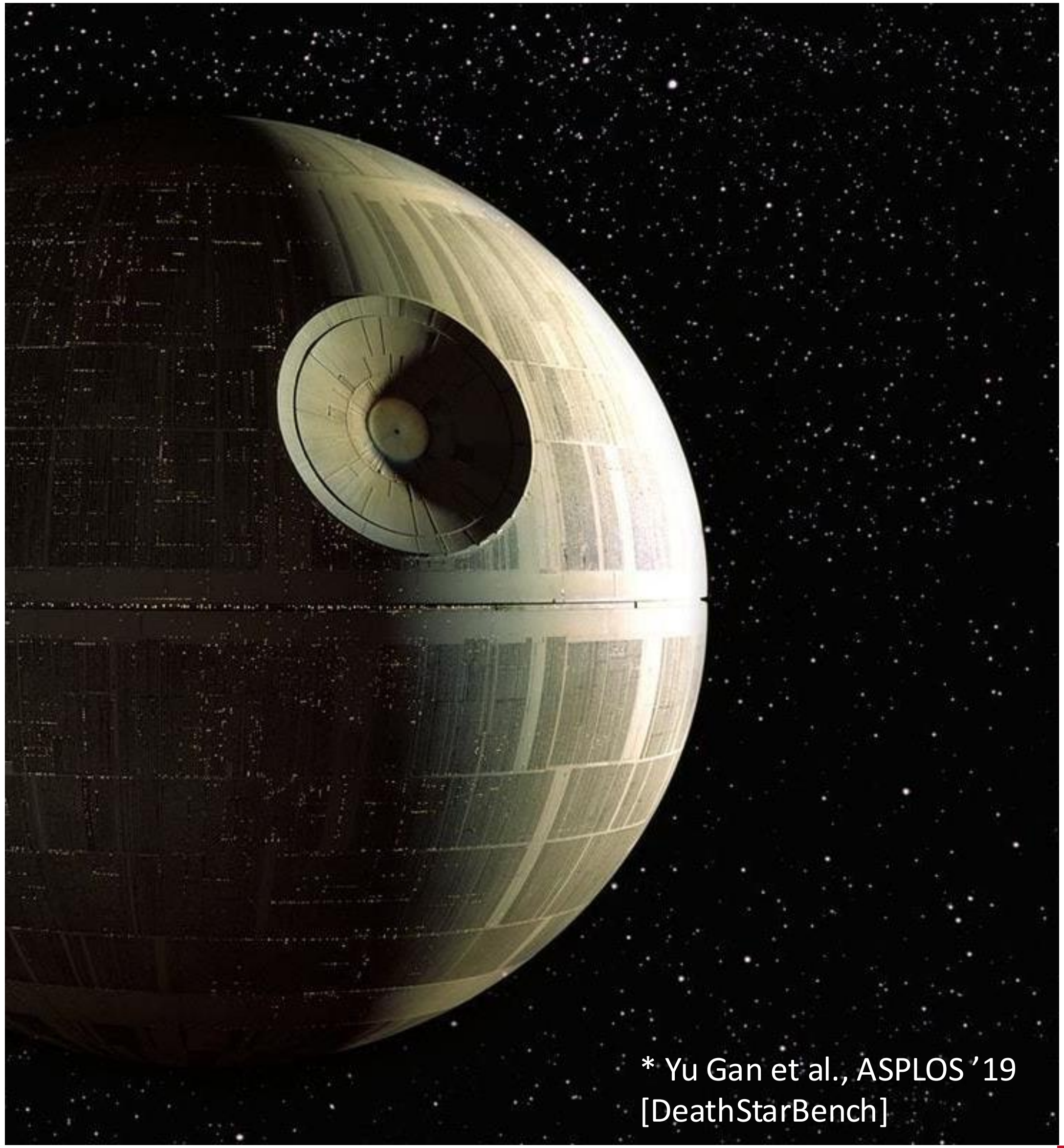
Fragmented into small units
(e.g., microservices)

New app paradigms: Ease of development, scalability, modularity...

Example: Web services built of microservices

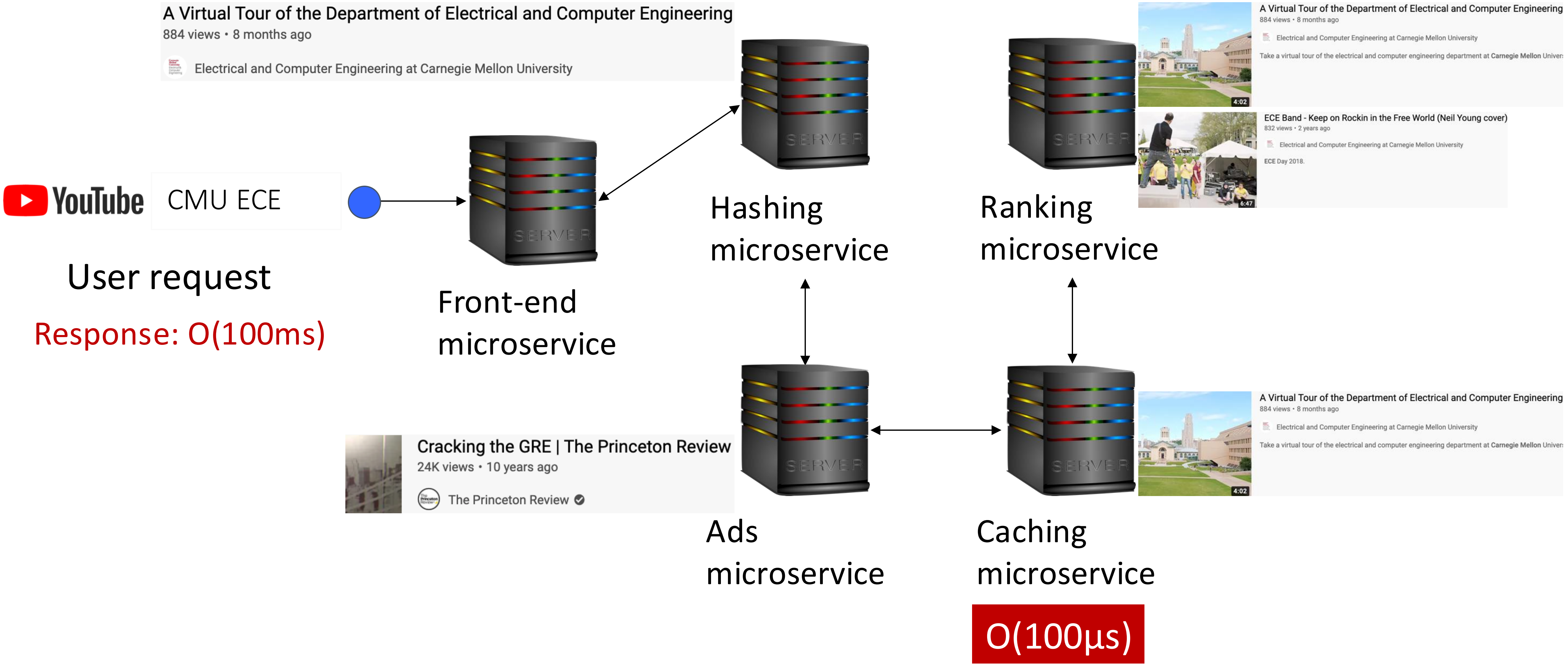






* Yu Gan et al., ASPLOS ’19
[DeathStarBench]

However: O(us) overheads matter for microservices



Must identify and mitigate us-scale overheads

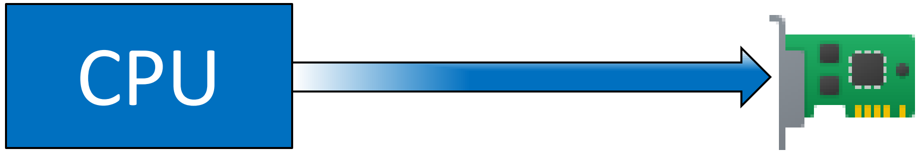
Carnegie Mellon University

Agenda

- INTRODUCTION TO DATA CENTER COMPUTING
- TREND 1: DECLINE IN HARDWARE PERFORMANCE SCALING
- TREND 2: GROWING DATA MOVEMENT OVERHEADS
- TRENDS 3, 4, 5: RAPID GROWTH IN DATA, USERS, FUNCTIONALITY
- **TREND 6: DISCONNECT BETWEEN CPU & I/O**
- TREND 7: A COMPLEX DESIGN SPACE

Radical Shift in Hyperscale Computing

Trend 6: Disconnect between CPU and I/O

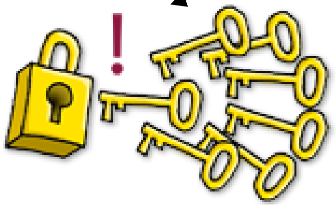


OS & legacy network stacks:

Microsecond-scale accesses

OS & legacy network stacks:

Microsecond-scale overheads



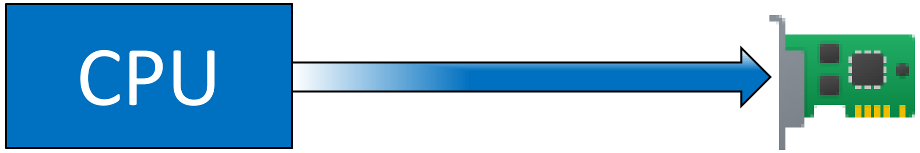
Thread contention

Thread wakeups

Context switches

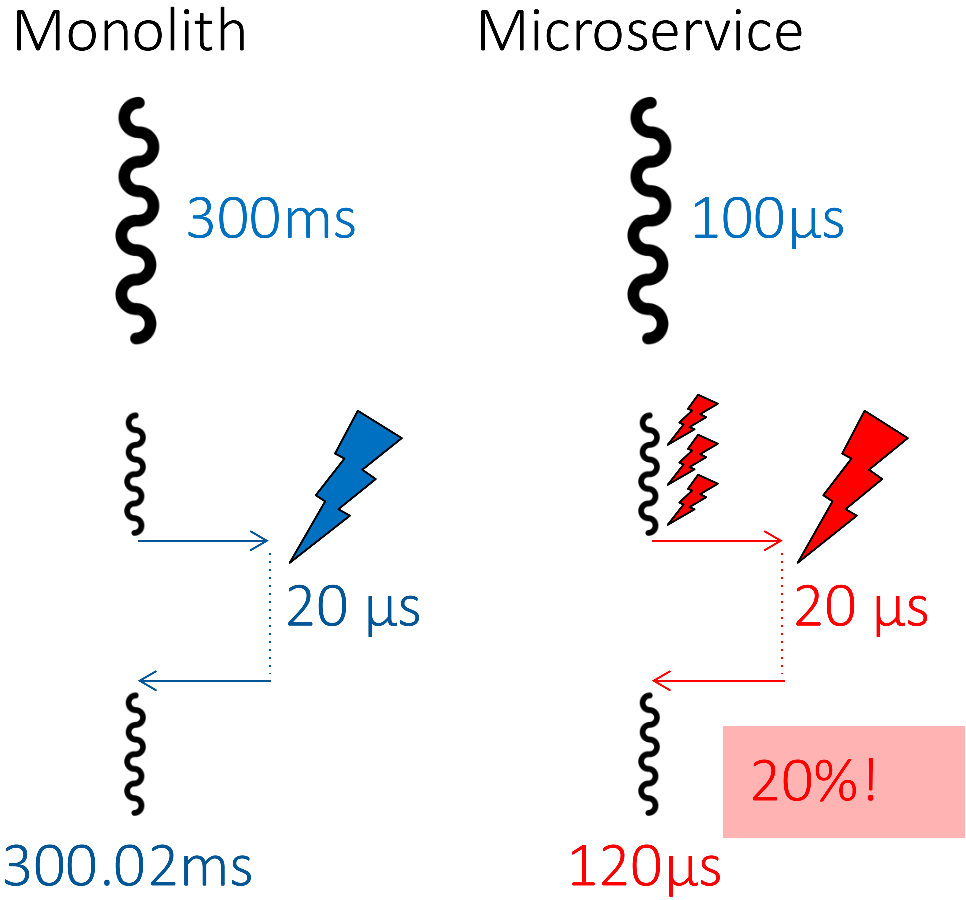
Radical Shift in Hyperscale Computing

Trend 6: Disconnect between CPU and I/O



OS & legacy network stacks:

Microsecond-scale accesses

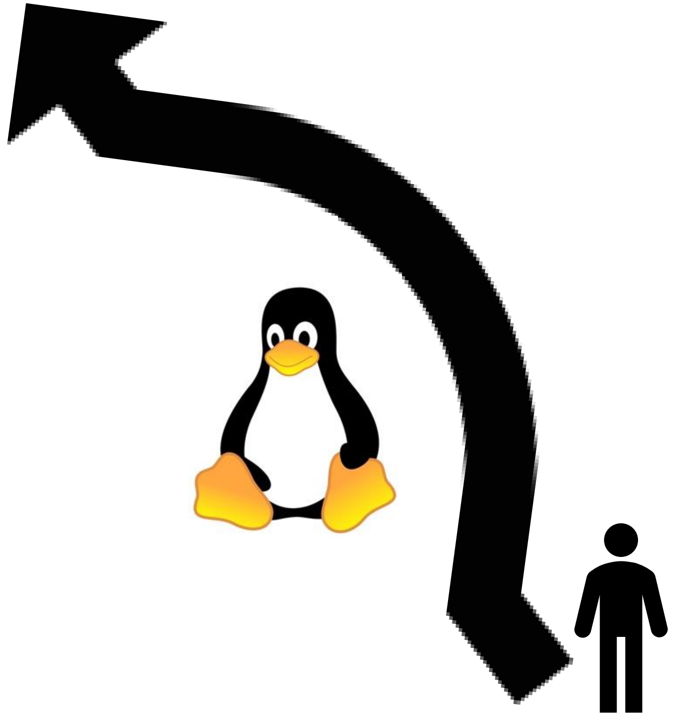


Challenge: OS & legacy software stacks have become the bottleneck

Trend 6: Disconnect between CPU and I/O

How has systems research adapted?

OS & Software Stacks:



Light-weight software stacks
(e.g., kernel bypass)

What are the challenges with this approach?

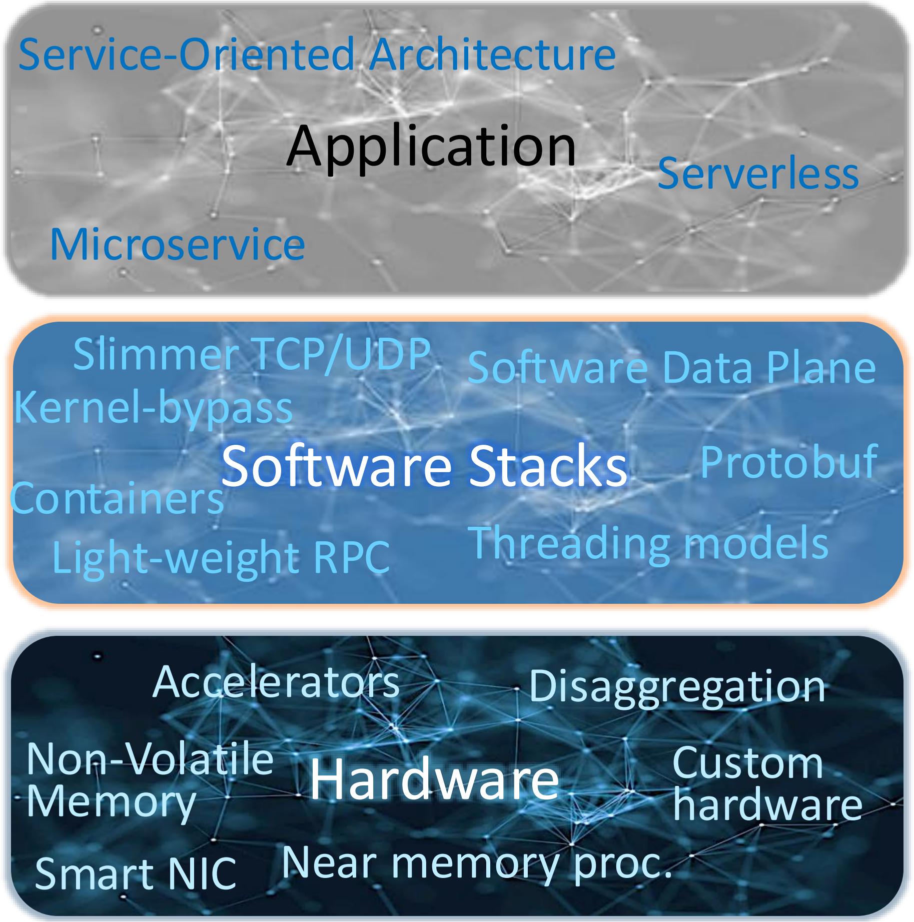
Carnegie Mellon University

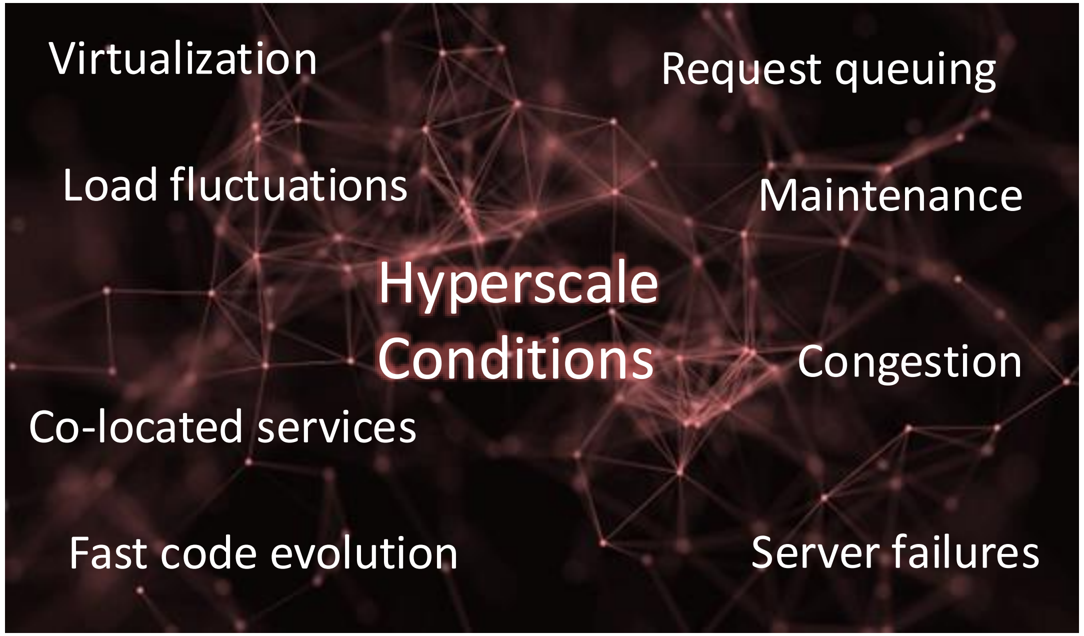
Agenda

- INTRODUCTION TO DATA CENTER COMPUTING
- TREND 1: DECLINE IN HARDWARE PERFORMANCE SCALING
- TREND 2: GROWING DATA MOVEMENT OVERHEADS
- TRENDS 3, 4, 5: RAPID GROWTH IN DATA, USERS, FUNCTIONALITY
- TREND 6: DISCONNECT BETWEEN CPU & I/O
- **TREND 7: A COMPLEX DESIGN SPACE**

Radical Shift in Hyperscale Computing

Trend 7: The paradox of choice - a complex design space



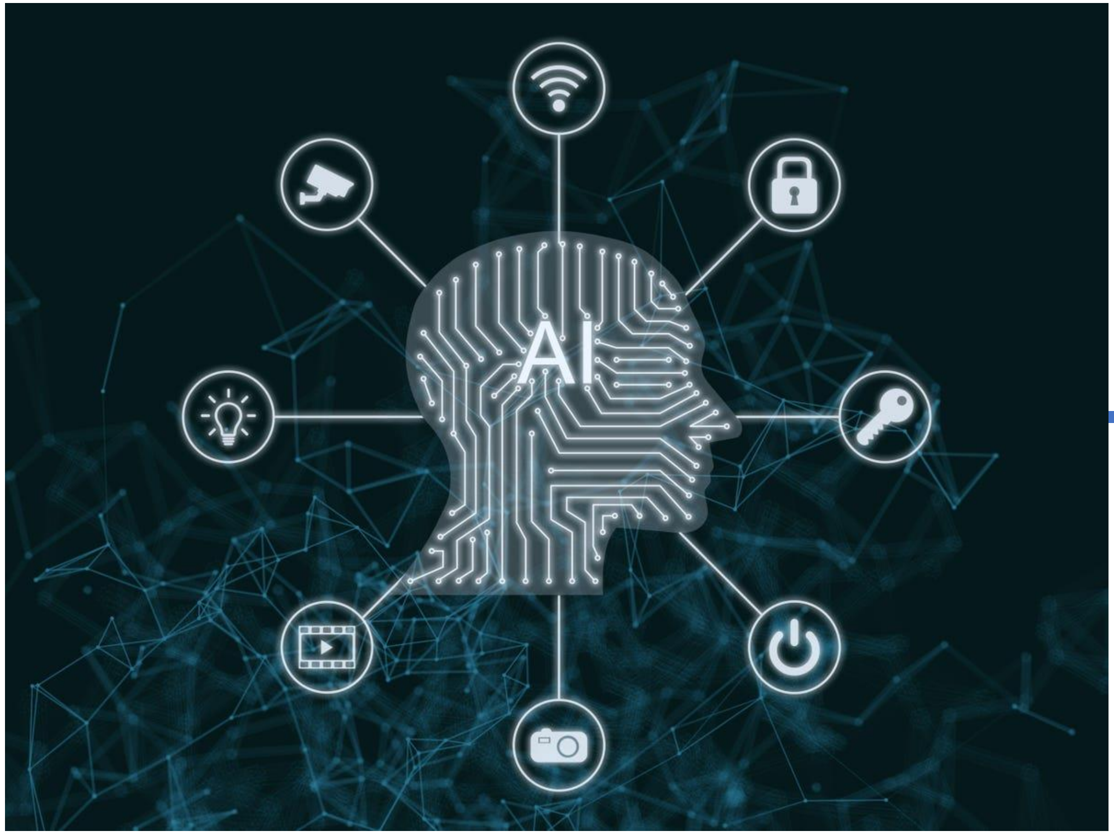


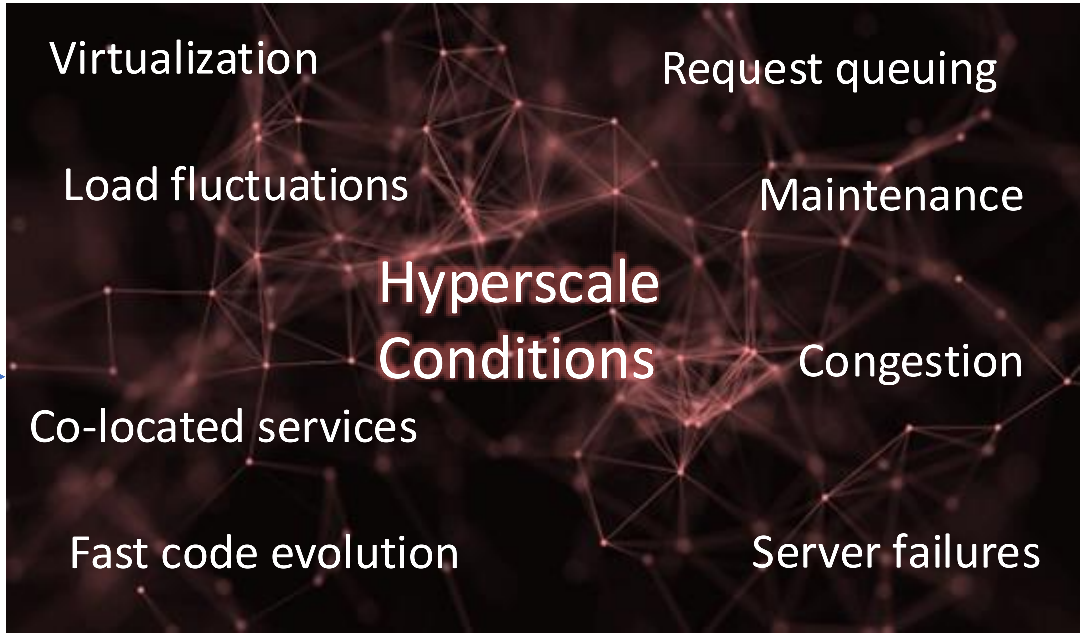
- Accelerators -> Threading -> Microservice -> Load
- Threading paradigms -> Service paradigms -> Load

…….

Trend 7: The paradox of choice - a complex design space

How has systems research adapted?





ML for system design

Data centers today

18-747: Data Center Computing

Akshitha Sriraman

Carnegie Mellon University

What are some “cool” emerging applications that you anticipate will grow to require data center support?

What are some metrics (other than performance, cost, energy) that large-scale systems might care about?

If you were designing the next Google/Facebook, what aspects of your data center would you design differently? Why?