COMP429/529 Parallel Programming

Didem Unat
dunat@ku.edu.tr

Lecture 1: Overview

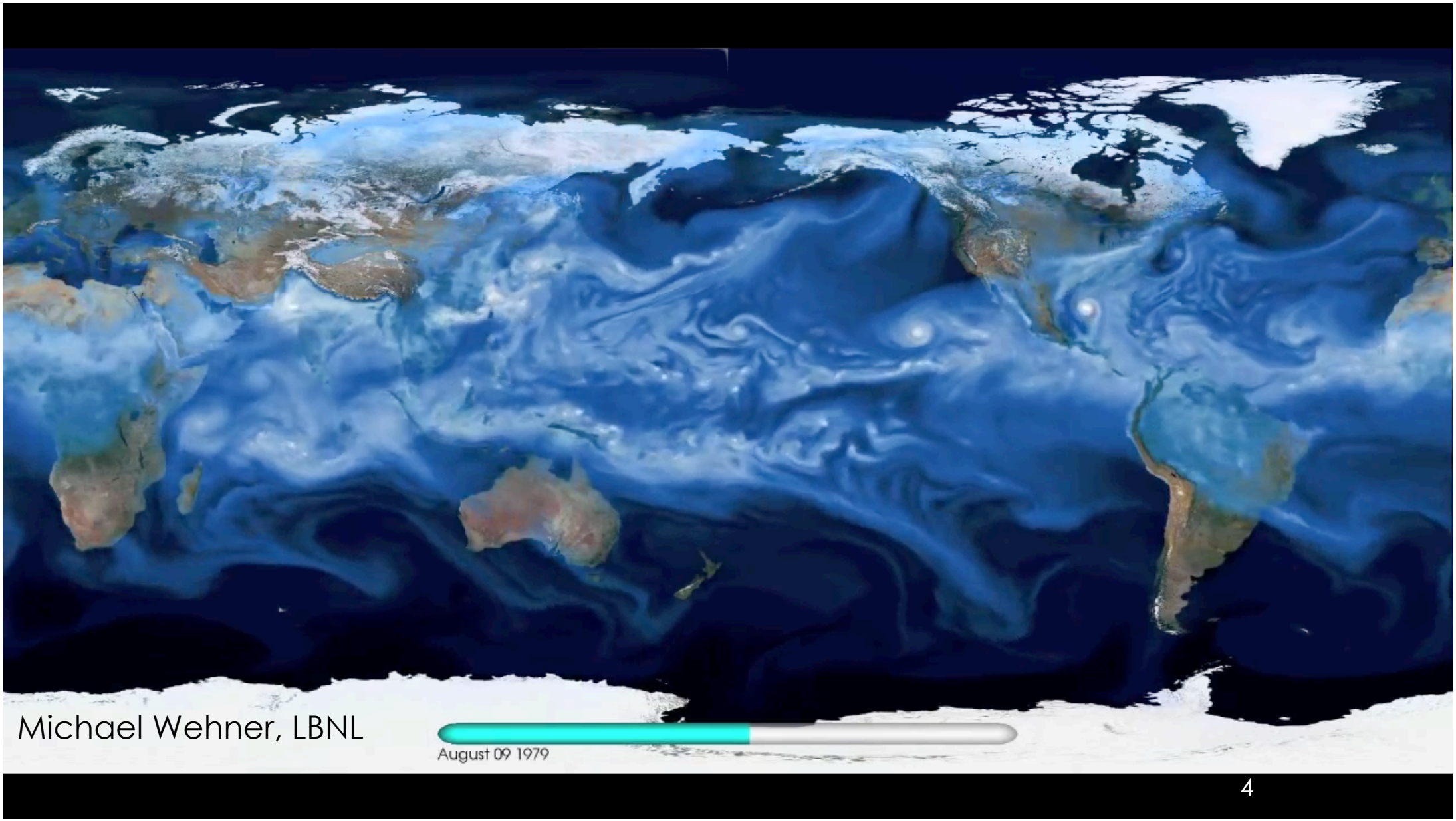
Motivation

- Why do we need parallel computers?

- Why study Parallel Programming?

Why do we need parallel computers?

- For higher performance
  - Fast execution
- In fact parallel computing is an *OLD concept*
  - The history goes back to the 1960s
  - Specialized computers (*supercomputers*)
- Scientific and Industrial Applications
  - Requires high performance capability
- Big Data and Deep Learning
  - Use parallel computers and parallel software
- Technology Trends
  - Because we have no choice ☺
  - Parallel architectures are everywhere


Michael Wehner, LBNL
August 09 1979

Scientific and Industrial Problems

- **Climate Change:** Understand the effects of global warming

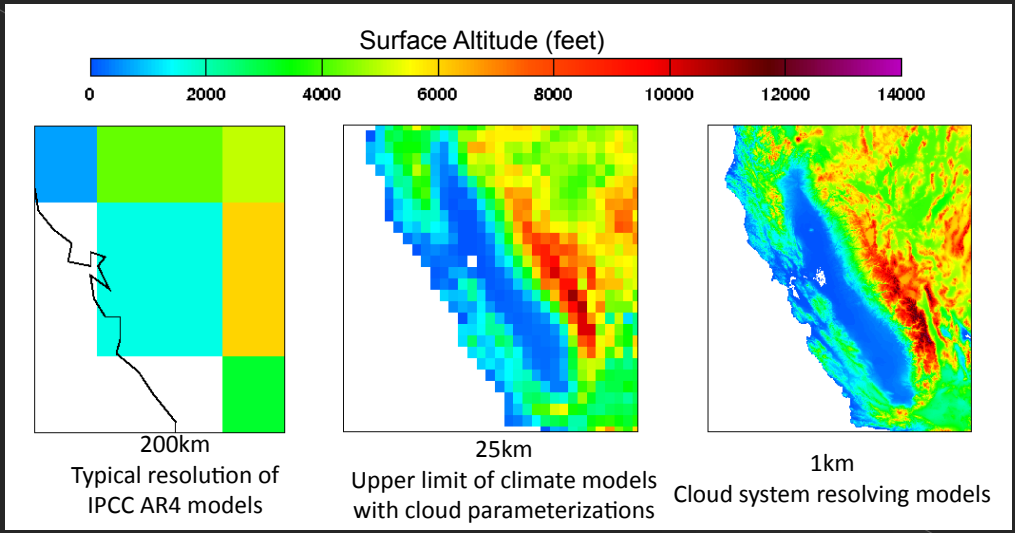


Image Source: Bill Collins

1 km

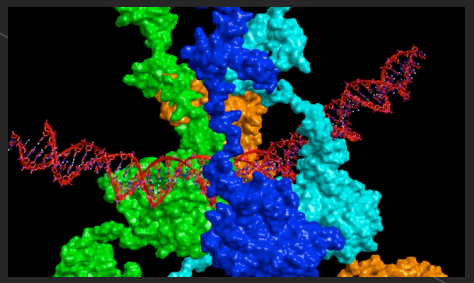
Multi-physics

More years/day

50-100 runs

- Many more simulations require high performance computing capability
- https://www.youtube.com/watch?v=l8M901ft20c&feature=youtu.be
- https://www.youtube.com/watch?v=9f-1-5BTNjQ

Some Other Application Areas



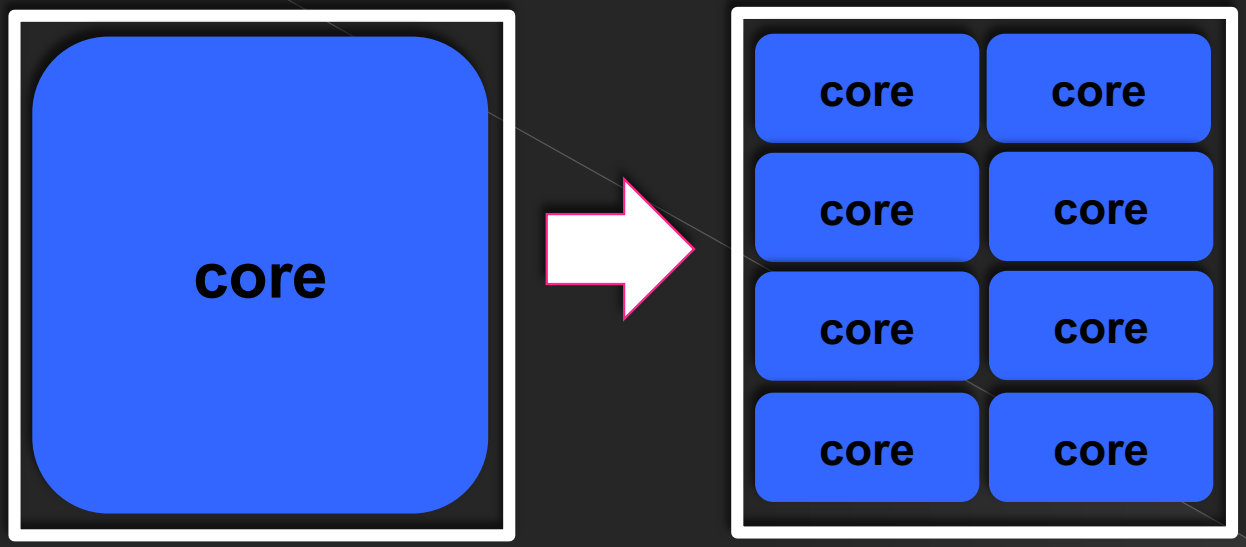
Drug Design (3M candidates)

Oil Exploration ($267.7)

Automobile design (60->24 months)

Data Analytics ($200B, 2017)

1st Trend in Computer Architecture

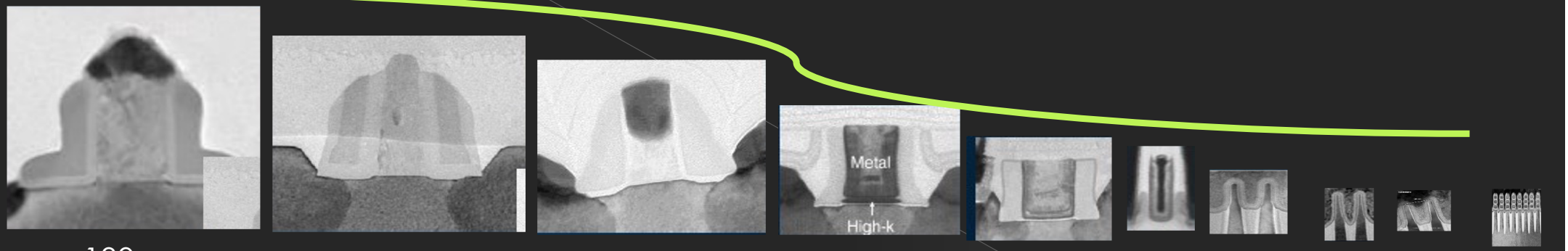


Massive on-chip parallelism

Moore's Law: 2X transistors / 18-24 months

2001- today

End of Moore's Law



Moore's law is the observation that the number of transistors in an integrated circuit doubles about every 2 years.

[1]: Marc Horowitz, Computing's Energy Problem (and what we can do about it), ISSC 2014, plenary
[2]: Moore: Landauer Limit Demonstrated, IEEE Spectrum 2012

End of Single Processor Era

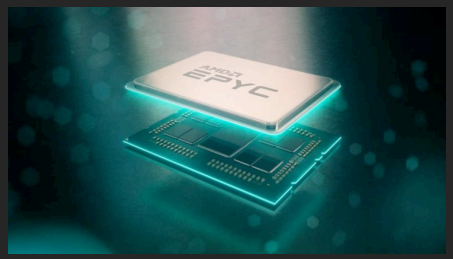
- Reinterpretation of Moore's law
  - No clock increases → multiple "cores" per chip
- Large number of cores on a chip
  - Already have 72-core chips (intel's Knights Landing)
  - 260 cores per processor in Sunway Supercomputer
- 1 Million to 1 Billion-way system level parallelism

Requires parallel programming

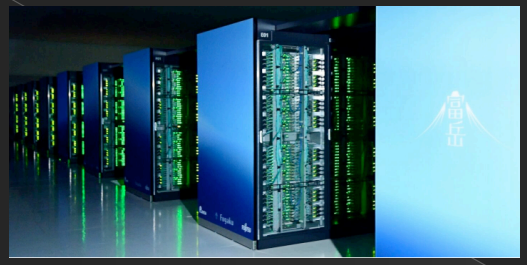
Parallel Computers are Everywhere

Quad-core Smart TV

Six-core A17

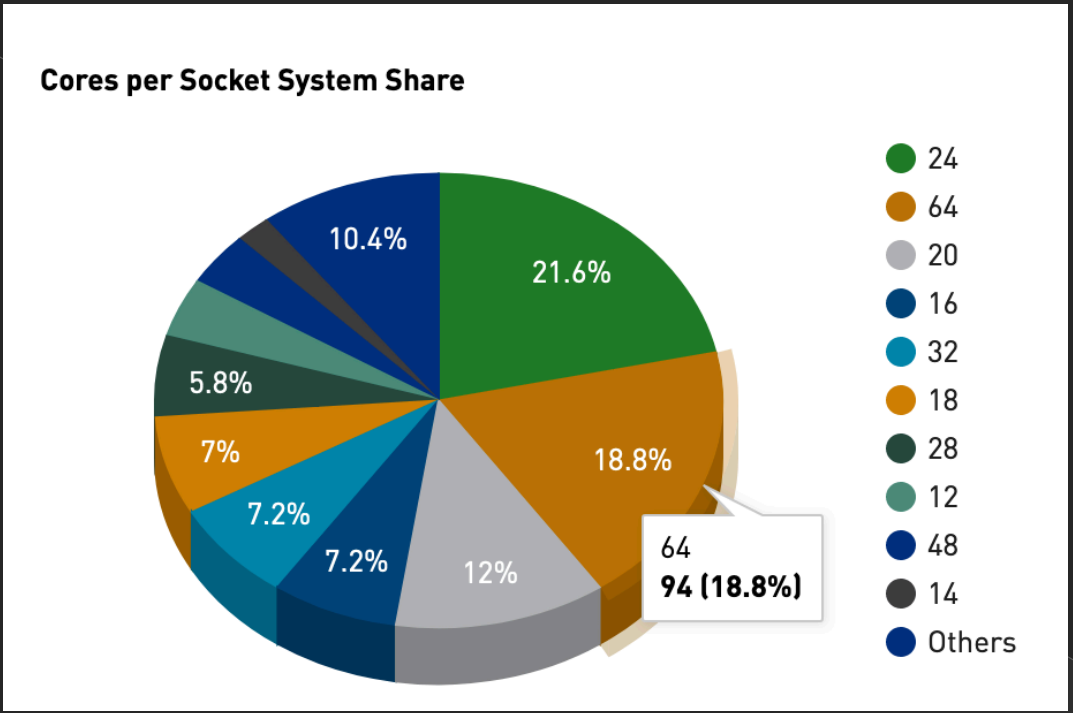


64-core AMD Epyc Milan



Fugaku 7.6M cores

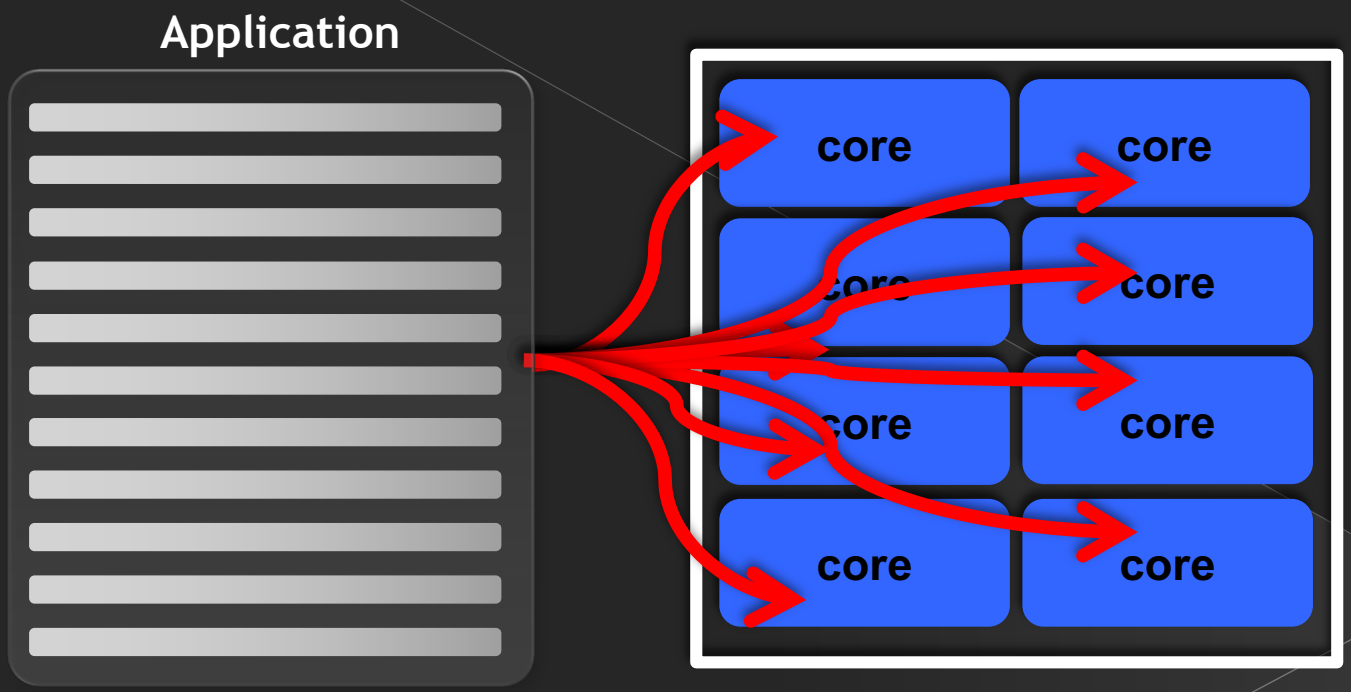
Number of Cores



- In Top 500 systems, all are multicore, 94 of the systems contain 64 cores per socket https://www.top500.org/statistics/list/

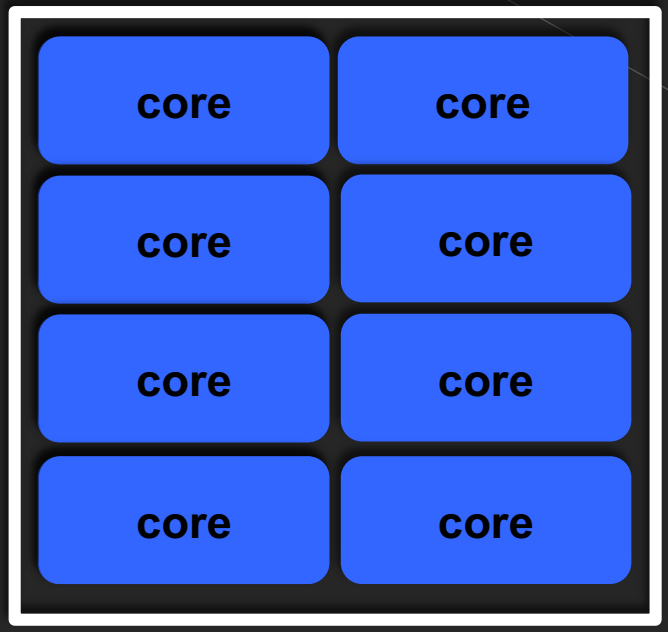
Multicore Programming

- Applications need to be restructured to take advantage of multiple cores on a chip



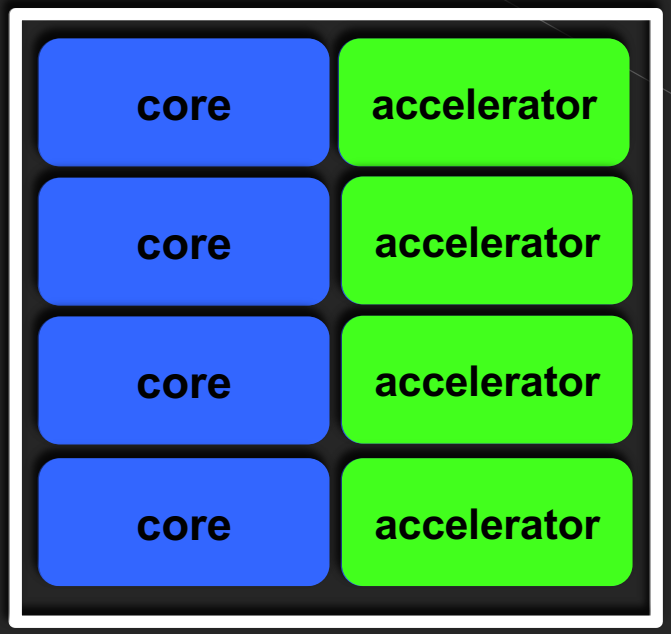
2nd Trend in Computer Architecture

heterogeneity



2nd Trend in Computer Architecture

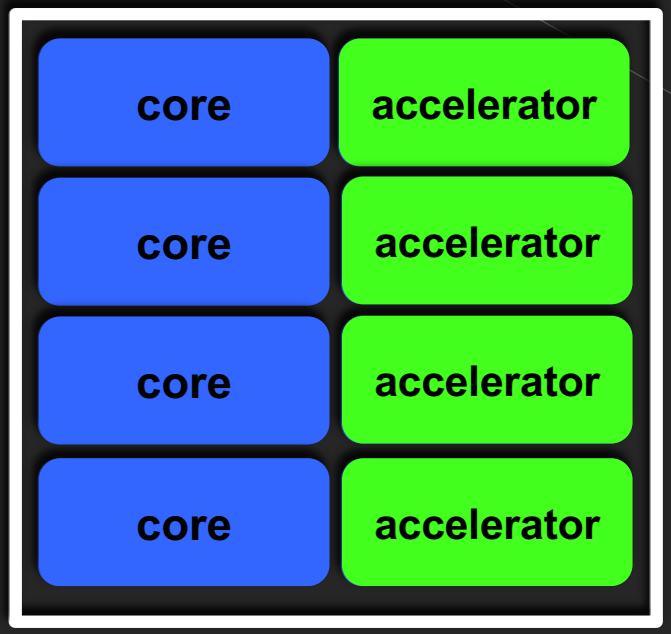
heterogeneity

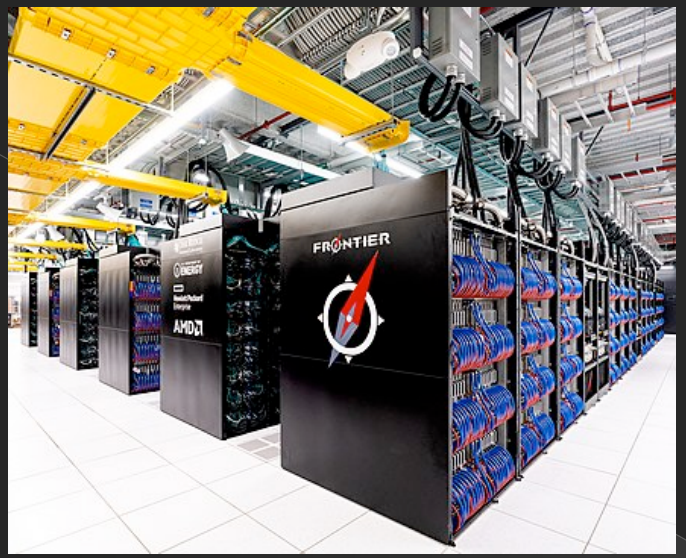


2nd Trend in Computer Architecture

heterogeneity

world's first and fastest
exascale supercomputer



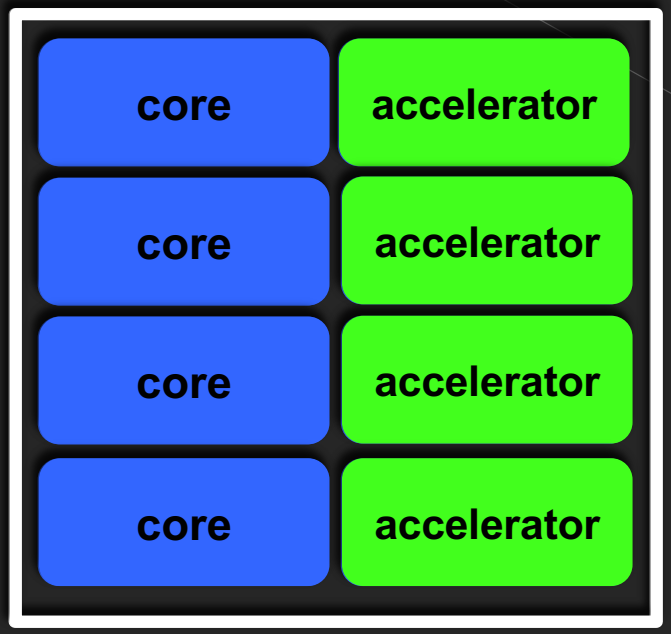


64 core AMD 3rd Gen Epyc
4 AMD Instinct 250X GPUs

2nd Trend in Computer Architecture

heterogeneity

world's first and fastest
exascale supercomputer

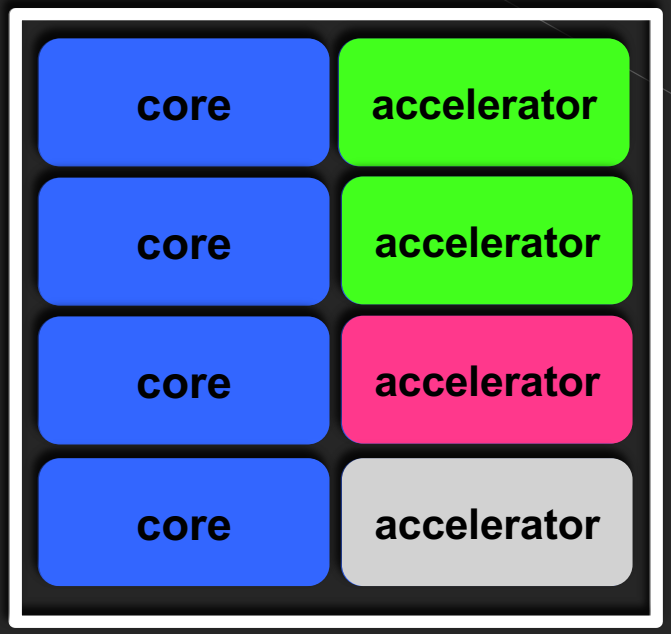


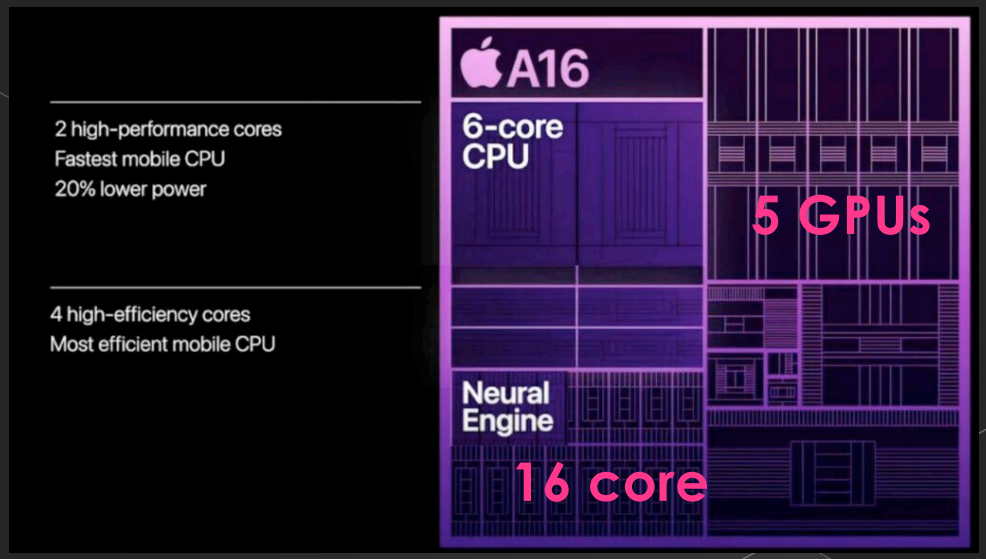
- **Frontier uses 9,472 nodes**
- **21 megawatts (MW)**
- **cost of US$600 million.**

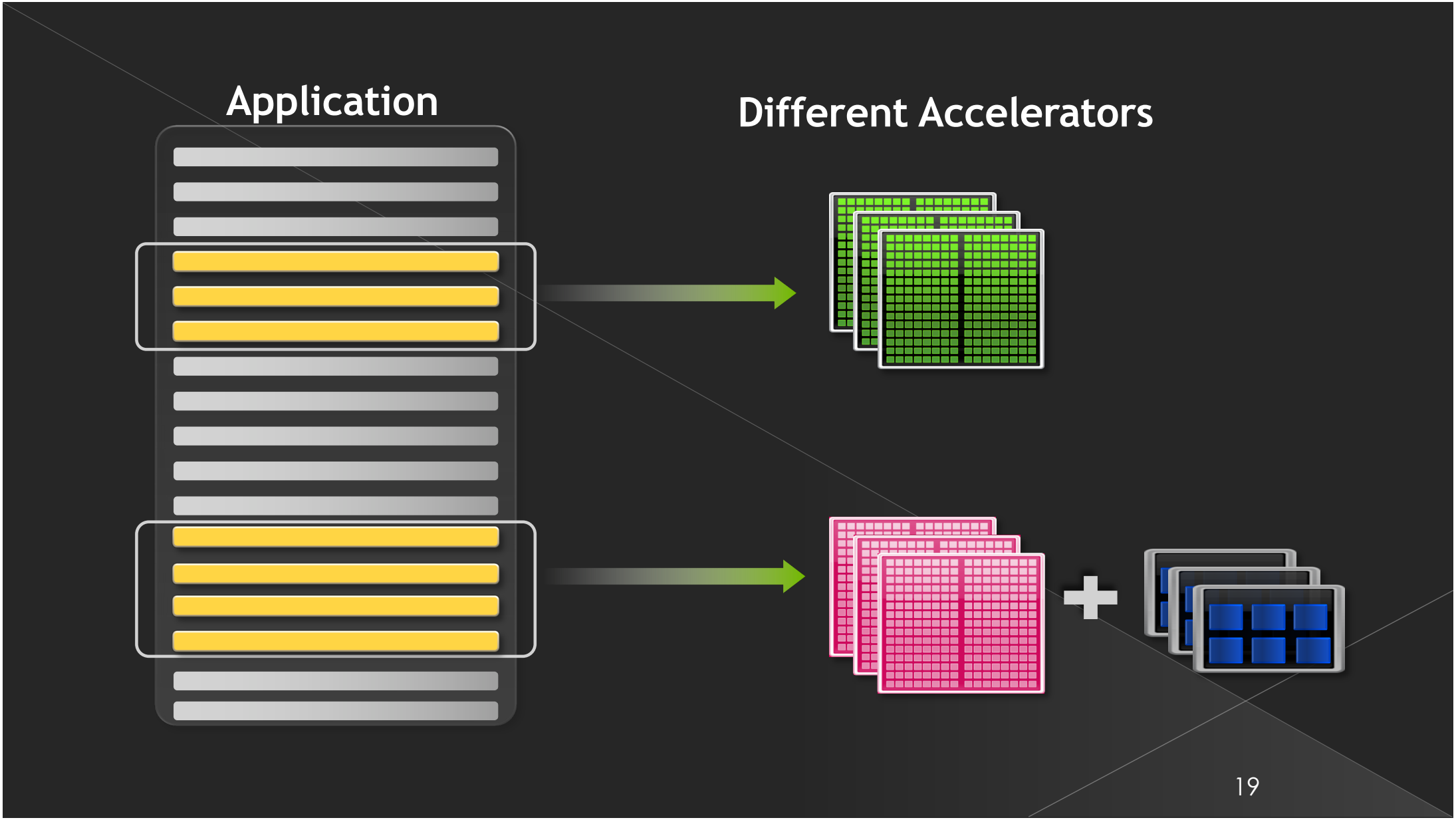
64 core AMD 3rd Gen Epyc
4 AMD Instinct 250X GPUs

2nd Trend in Computer Architecture

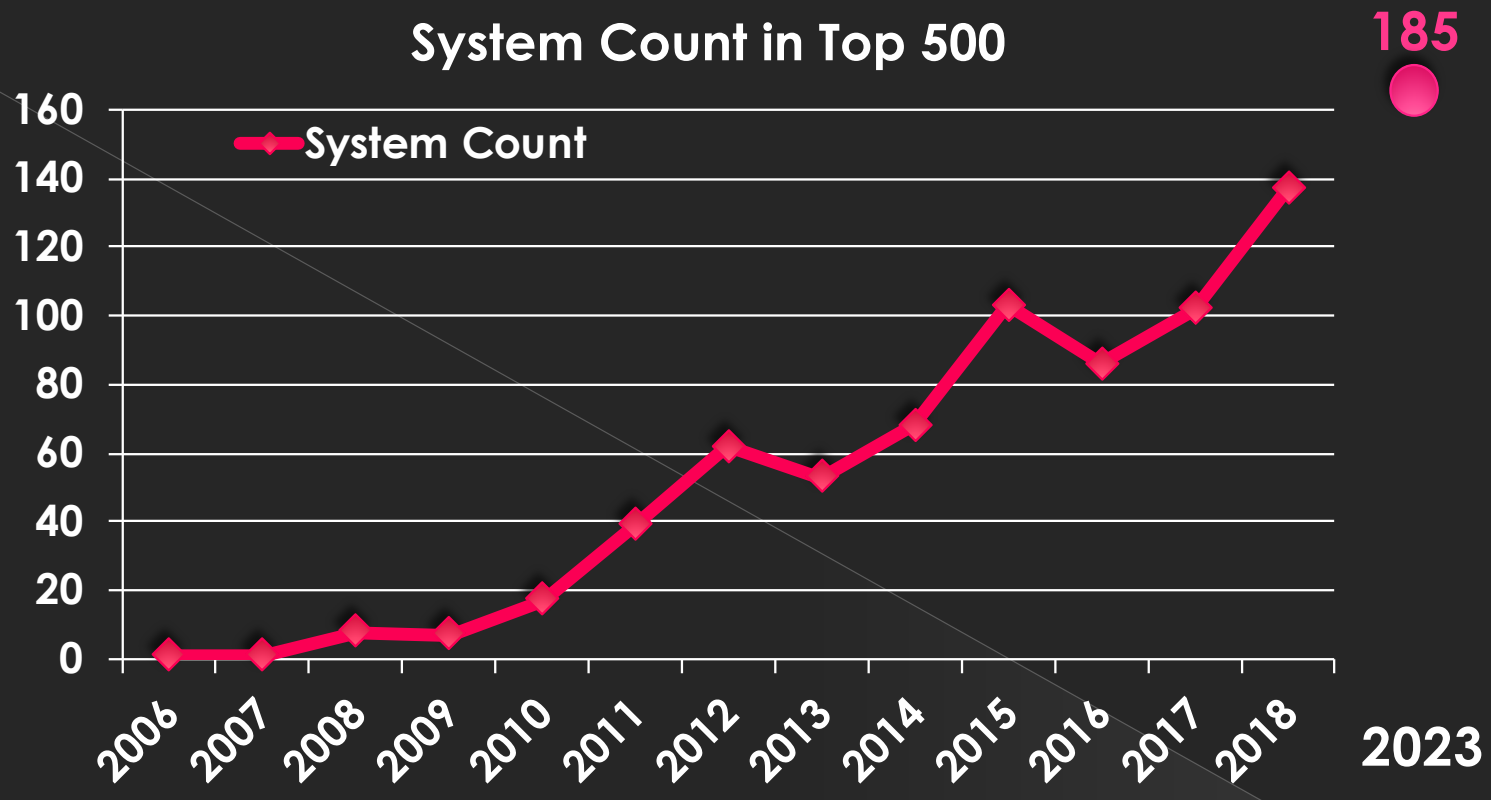
heterogeneity





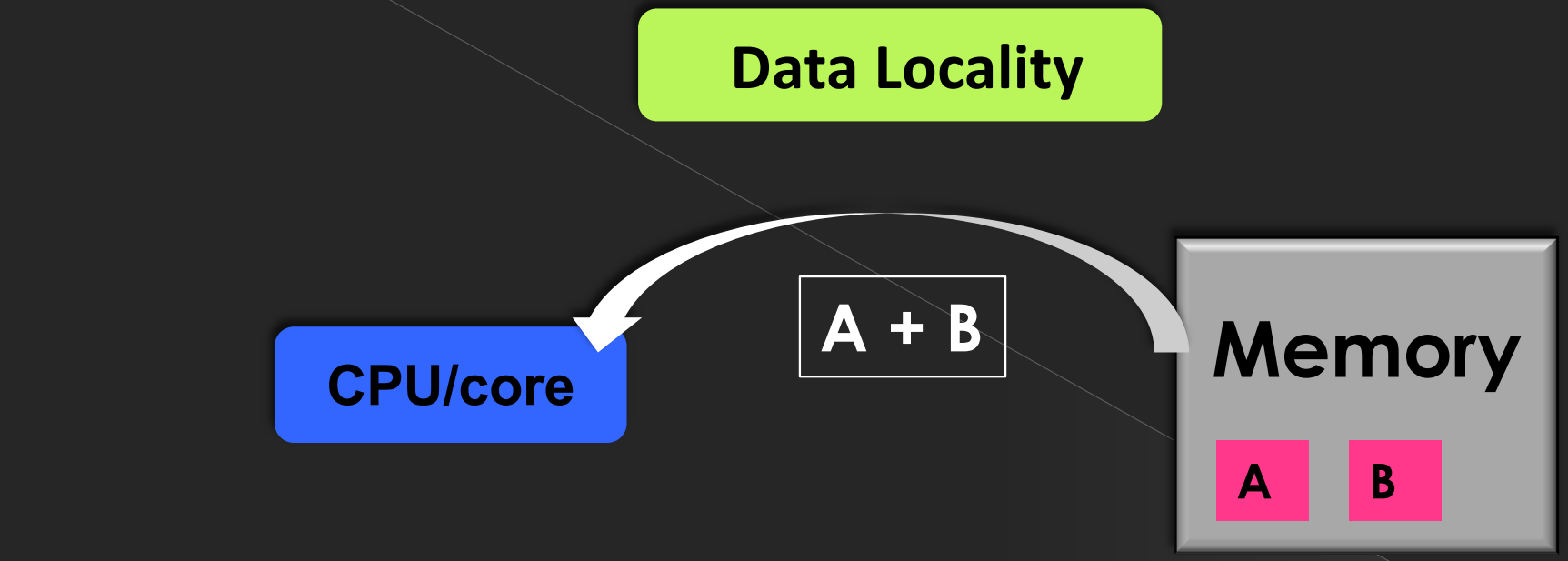

Application
Different Accelerators

Heterogeneous Systems in Top 500



KUACC on our campus has +100 GPUs (accelerators).

3rd Trend in Computer Architecture



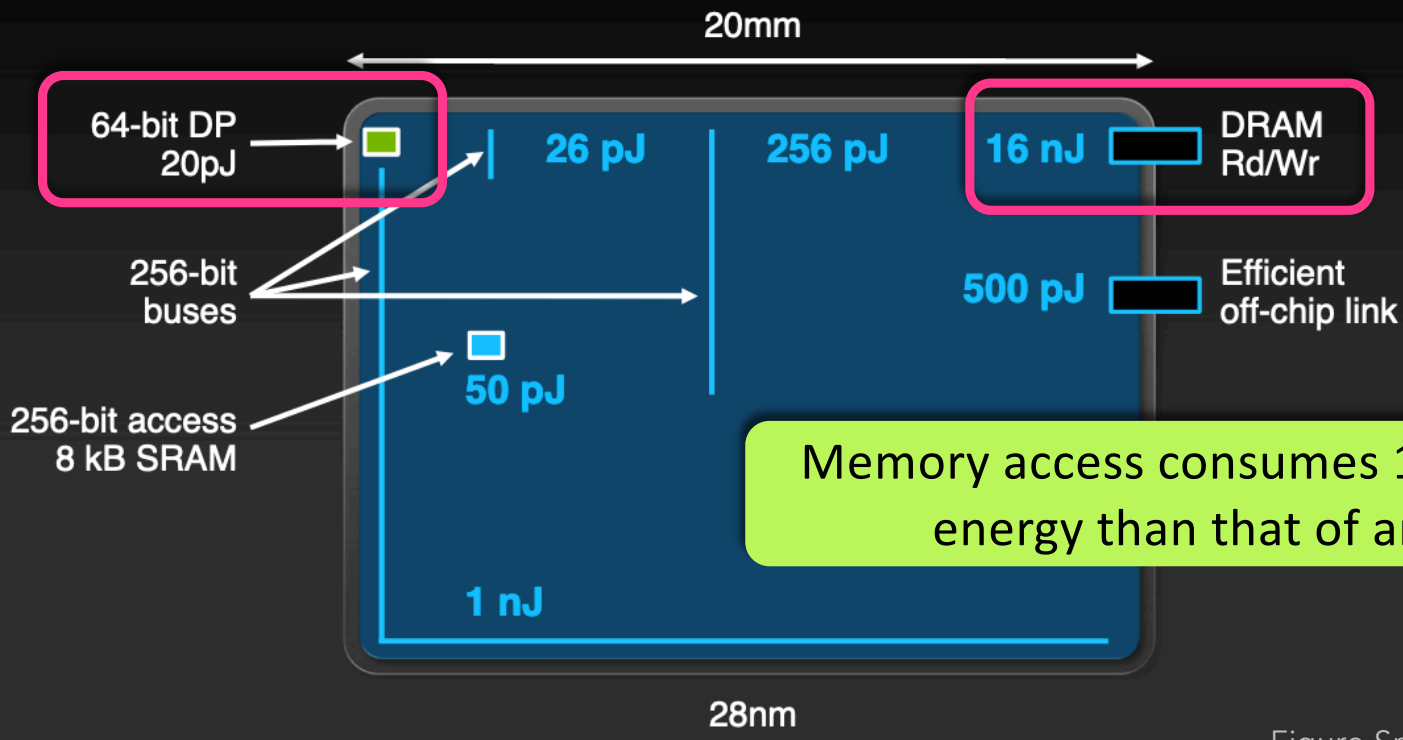
Computing is cheap!

Data movement is expensive!

[1]: Marc Horowitz, Computing's Energy Problem (and what we can do about it), ISSC 2014, plenary

The High Cost of Data Movement

Fetching operands costs more than computing on them



Memory access consumes 100-1000x more energy than that of an addition

Figure Src: Bill Dally, SC10

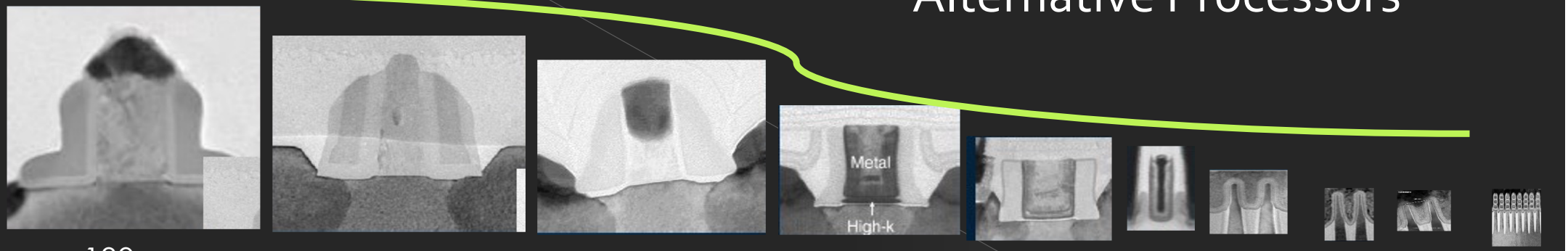
Efficient Computing = Data Locality

4th Trend in Computer Architecture

End of Moore's Law

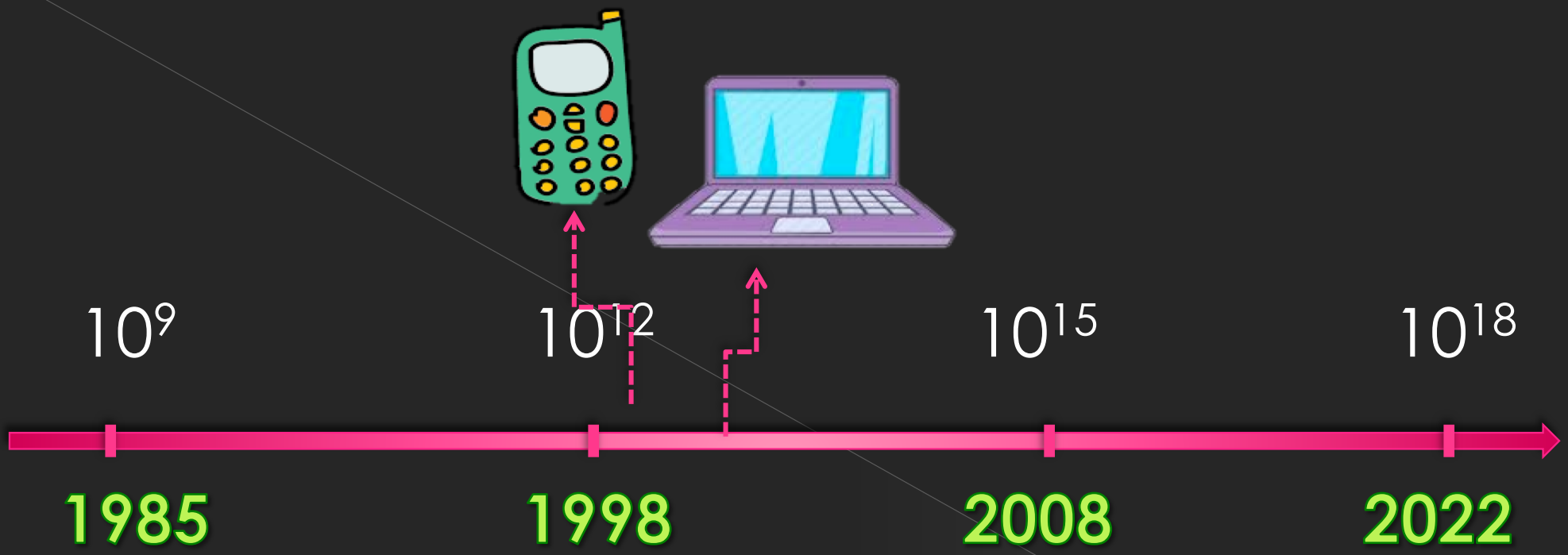
2001- today

Alternative Processors



- Cost of manufacturing [3]
  - $170M for a 10 nm chip, $300M for a 7 nm chip, $500M for a 5 nm chip
  - For specialized chips, even higher

[1]: Marc Horowitz, Computing's Energy Problem (and what we can do about it), ISSC 2014, plenary
[2]: Moore: Landauer Limit Demonstrated, IEEE Spectrum 2012


10^9
10^{12}
10^{15}
10^{18}
1985
1998
2008
2022

Performance

- This course is about ``Performance''
- In this class, we are not only interested in solving a problem in **parallel**, but also developing **efficient** solutions.

Parallelism | Heterogeneity | Data Locality

Course Basics

- Website
  - Blackboard : https://ku.blackboard.com/
  - All course materials will be posted (hopefully before the class)
- Main Book
  - An Introduction to Parallel Programming by Peter Pacheco (ISBN: 978-0-12-374260-5)
- *Additional reading materials will be posted on blackboard as needed*

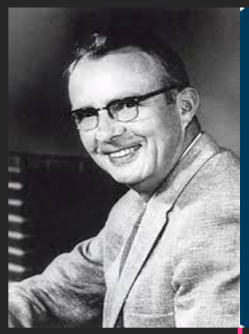
Didem Unat?

2006
Graduated from
Boğaziçi University

2012
PhD at
University of California,
San Diego

Didem Unat?

TEDx Istanbul
x = independently organized TED event



BERKELEY LAB
Lawrence Berkeley National Laboratory

MARIE CURIE ACTIONS

BİLİM AKADEMİSİ

THE ROYAL SOCIETY

2012-2014
Luis Alvarez Postdoctoral Fellowship

Lawrence Berkeley National Laboratory

2014 -
Koç University

KOÇ UNIVERSITY
PARALLEL AND MULTICORE COMPUTING LABORATORY

Didem Unat?

SPAR CITY

beyond moore

2.6M Euro
6 European partners

EuroHPC Joint Undertaking

1.5M Euro
First in Turkey
in Computer Eng.

erc
European Research Council
Established by the European Commission

COMMUNICATIONS OF THE ACM

Koc University

HOME | CURRENT ISSUE | NEWS | BLOGS | OPINION | RESEARCH | PRACTICE | CAREERS | ARCHIVE

Home / Careers / Didem Unat Named SIGHPC Emerging Woman Leader in Technical... / Full Text

ACM CAREERS

Didem Unat Named SIGHPC Emerging Woman Leader in Technical Computing

September 8, 2021
Comments

VIEW AS: SHARE:

Didem Unat is an assistant professor of computer engineering at Ko University in Istanbul, Turkey.

Didem Unat of Koç University has been named the winner of the 2021 ACM SIGHPC Emerging Woman Leader in Technical Computing award. Unat was recognized for innovations in the field of programming models for data locality in high performance and scientific computing and for her leadership role in the international high performance computing community.

Unat's work on simplifying software development for current and future supercomputing architectures resulted in architecture-independent abstractions. These allow for the development of scientific software that maps to complex memory hierarchies and accelerator structures, with high-performance results.

"Unat's rigorous technical work has directly impacted the productivity of application scientists," says award committee chair Cristina Beldica of Intel. " This is critical not only for high performance computing, but science in general."

From *HPCwire*
View Full Article

SIGN IN for Full Access
User Name
Password
» Forgot Password?
» Create an ACM Web Accou
SIG

MORE NEWS & OPINIONS
Phone's Dark Mode Doesn
Necessarily Save Much Ba
Life
Tech Explorist
Lessons From the Loo
Lorrie Faith Cranor
Inclusive Integration of
Computing in School Distr
Two Essential Tradeoffs
Merijke Coenraad, Brian Giovanini,
Mills, Jeremy Roschelle

Requirements for the Class

- Basic knowledge in Computer Architecture
  - Memory, cache, core concepts
- **C Programming**
  - Pointers, dynamic memory allocation, multidim arrays, std input/output
    - http://www.cprogramming.com/tutorial/c-tutorial.html
- Basic Linux commands (Some OS background)
  - Remote access, transfer files to/from clusters
  - Develop and test programs on clusters
- Command line editors
  - Highly recommend getting familiar with Emacs or Vi
- Want to locally develop your programs?
  - Have to install parallel libraries locally
  - Still need to collect performance data on **campus clusters**

Access to Clusters on Campus

- You all will have an account on campus clusters
  - Namely KUACC (KU Advanced Computing Center)
  - https://kuacc.ku.edu.tr/
- Login with your KU username and password
- Need to be on campus network
  - If cannot login, send an email to hpc-support@ku.edu.tr and CC me.
- Off-campus connection requires VPN
  - Test at home, set up VPN connection by following the instructions at http://my.ku.edu.tr/
  - If you have a problem, open a ticket or ask help from your friend

TAs and Office Hours

- Ilyas Turimbetov (ENG 230)
- Kefah Issa (ENG 230)
- Office hours
  - Tu 5.30 - 7 pm
  - Th 1 – 2.30 pm

Grading

- 60 % (20 + 20 + 20) Three Programming Assignments
- 30 % (3 x 10%) In-class Quizzes (30 mins in-person)
- 10 % Participation

- **Submission due dates and times are *strictly enforced*, and no late submissions will be accepted.**
- **In order to receive a passing grade, you have to receive a non-zero score from each category above (assignments, exams, and participation).**
- **No make-up for quiz**

Programming Assignments

- Three assignments
  - (3 weeks each)
- These will take time (a lot of time)
  - Do not wait for last minute, you won't be able to finish it
- Each assignment requires coding and writing a report
  - You will be evaluated based on your code, program's performance and report

Course Policy

- NO late arrival or early leave to/from the lecture

- **You must submit your own work in all assignments, projects and exams. Academic dishonesty includes using other people's words or ideas without acknowledgement, cheating on exams, projects, and homework.**
- **In case any of the academic dishonesties are disclosed, disciplinary action will follow**