

Future of Computing: Moore's Law & Its Implications

Introduction to Computer Systems
29th Lecture, Dec. 23, 2024

Instructors:

Class 1: Chen Xiangqun, Liu Xianhua

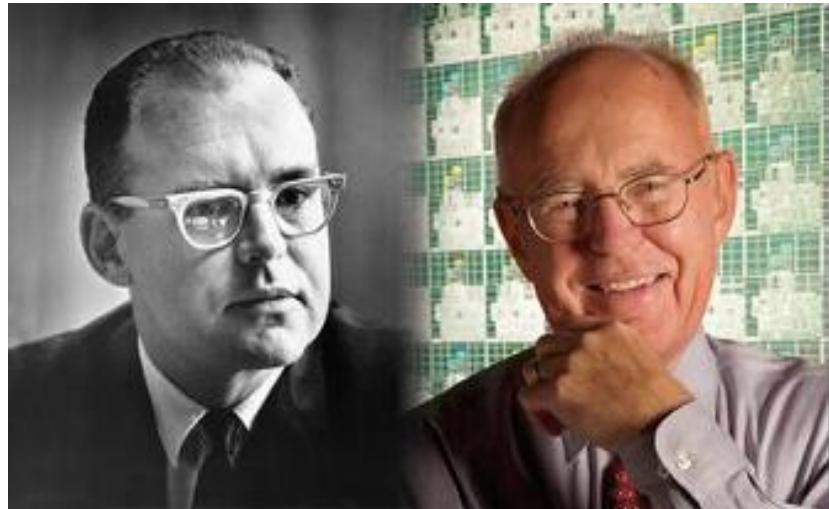
Class 2: Guan Xuetao

Class 3: Lu Junlin

Moore's Law Origins



April 19, 1965



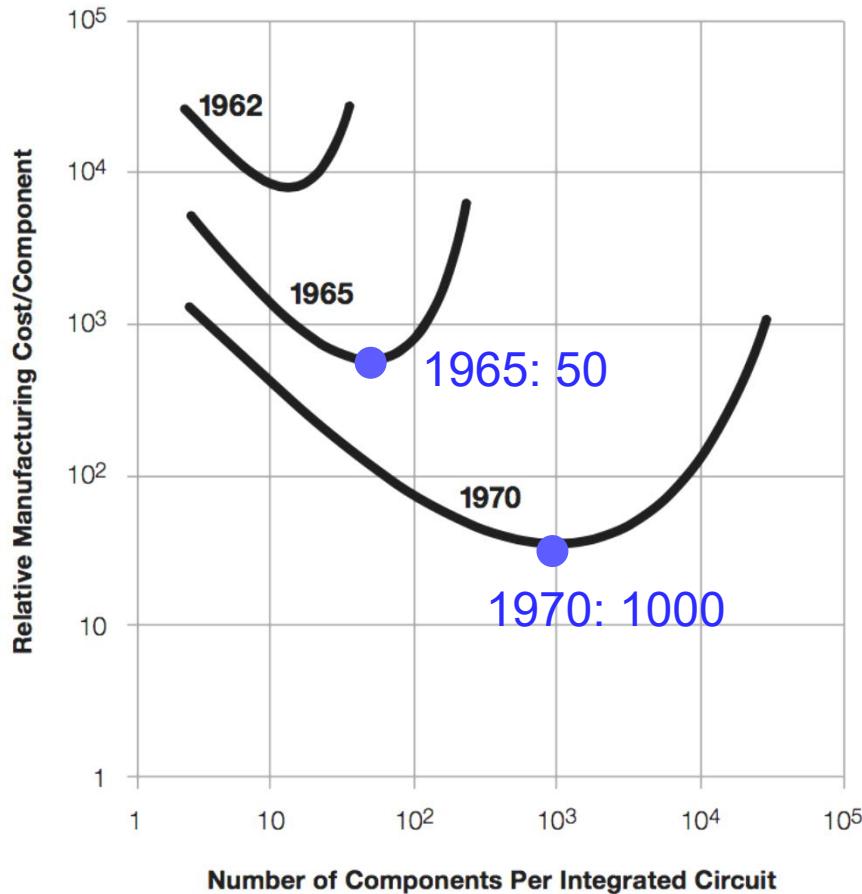
Cramming more components onto integrated circuits

With unit cost falling as the number of components per circuit rises, by 1975 economics may dictate squeezing as many as 65,000 components on a single silicon chip

By Gordon E. Moore

Director, Research and Development Laboratories, Fairchild Semiconductor
division of Fairchild Camera and Instrument Corp.

Moore's Law Origins

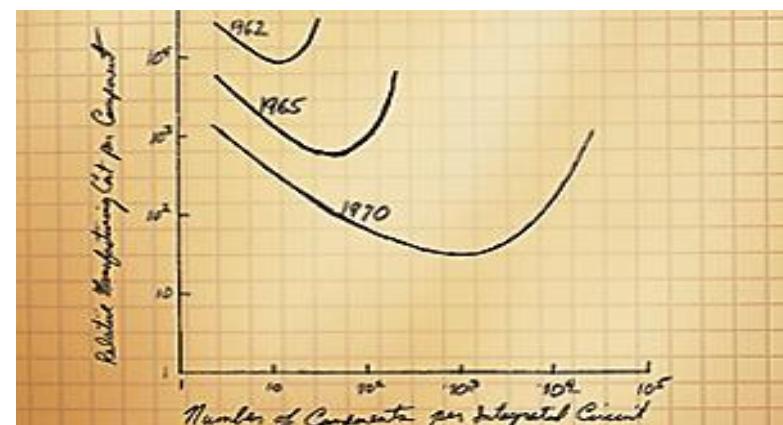


- **Moore's Thesis**

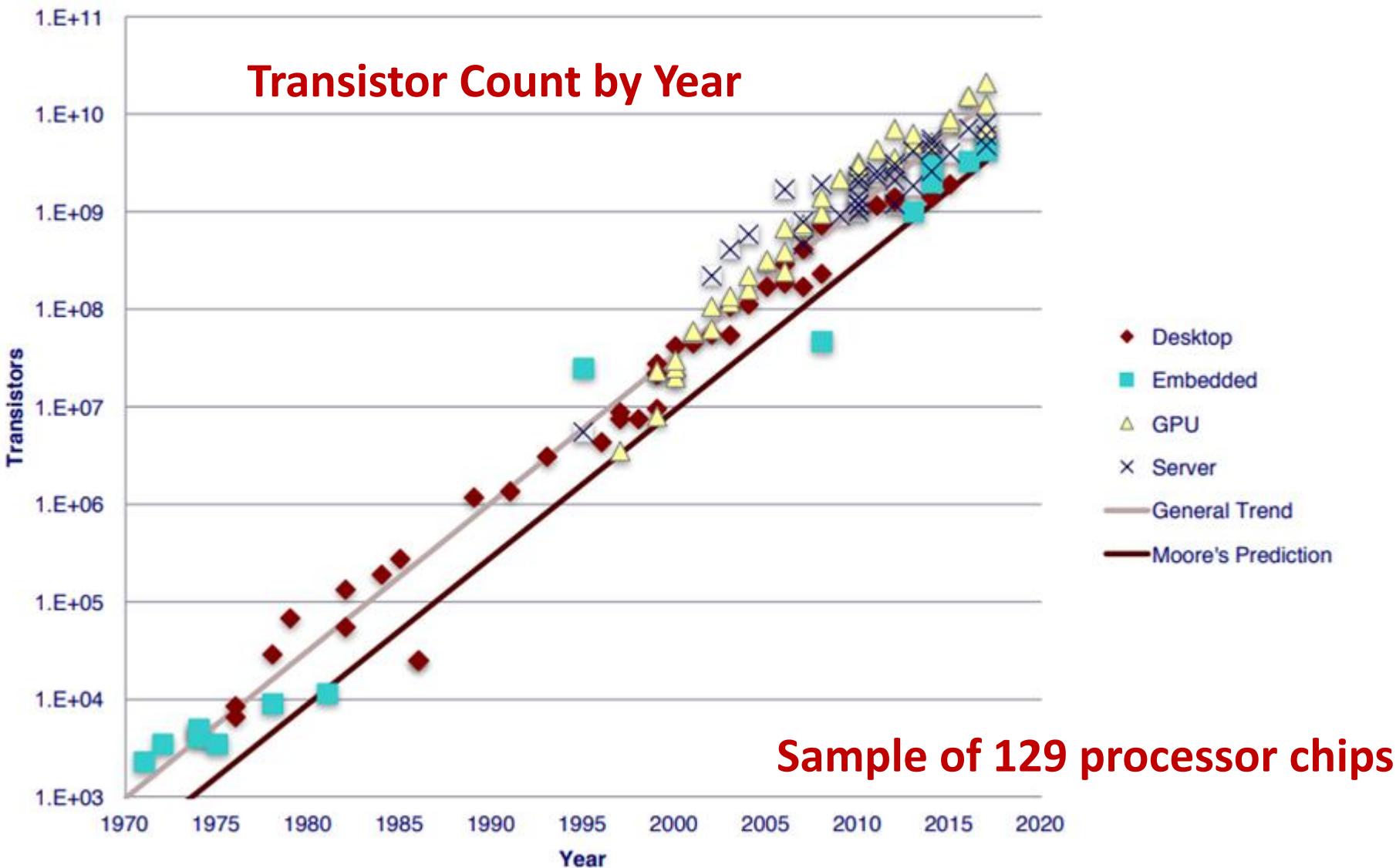
- Minimize price per device
- Optimum number of devices / chip increasing 2x / year

- **Later**

- 2x / 2 years
- “Moore’s Prediction”



Moore's Law: 50 Years



What Moore's Law Has Meant



○ 1976 Cray 1

- 250 M Ops/second
- ~170,000 chips
- 0.5B transistors
- 5,000 kg, 115 KW
- \$9M
- 80 manufactured



○ 202x Huawei Mate Pro Series

- > 10 B Ops/second
- 8 CPU + 20 GPU + 2 NPU
- > 15B transistors
- ~ 200g, < 5 W
- ~ \$800
- > 14 million sold

What Moore's Law Has Meant

- 1965 Consumer Product



transistor radio

- 2020 Consumer Product

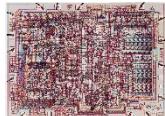


HiSilicon Kirin 9000 with
15.3B transistors

Visualizing Moore's Law to Date

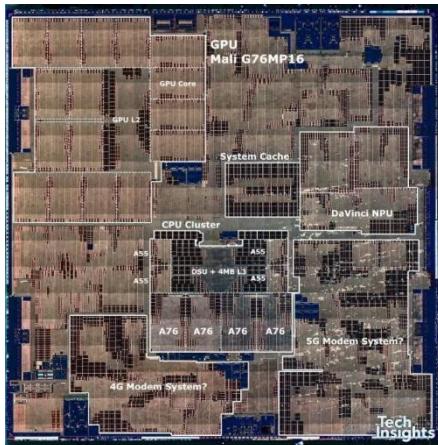
If transistors were the size of a grain of sand

Intel 4004
1971
2,300 transistors



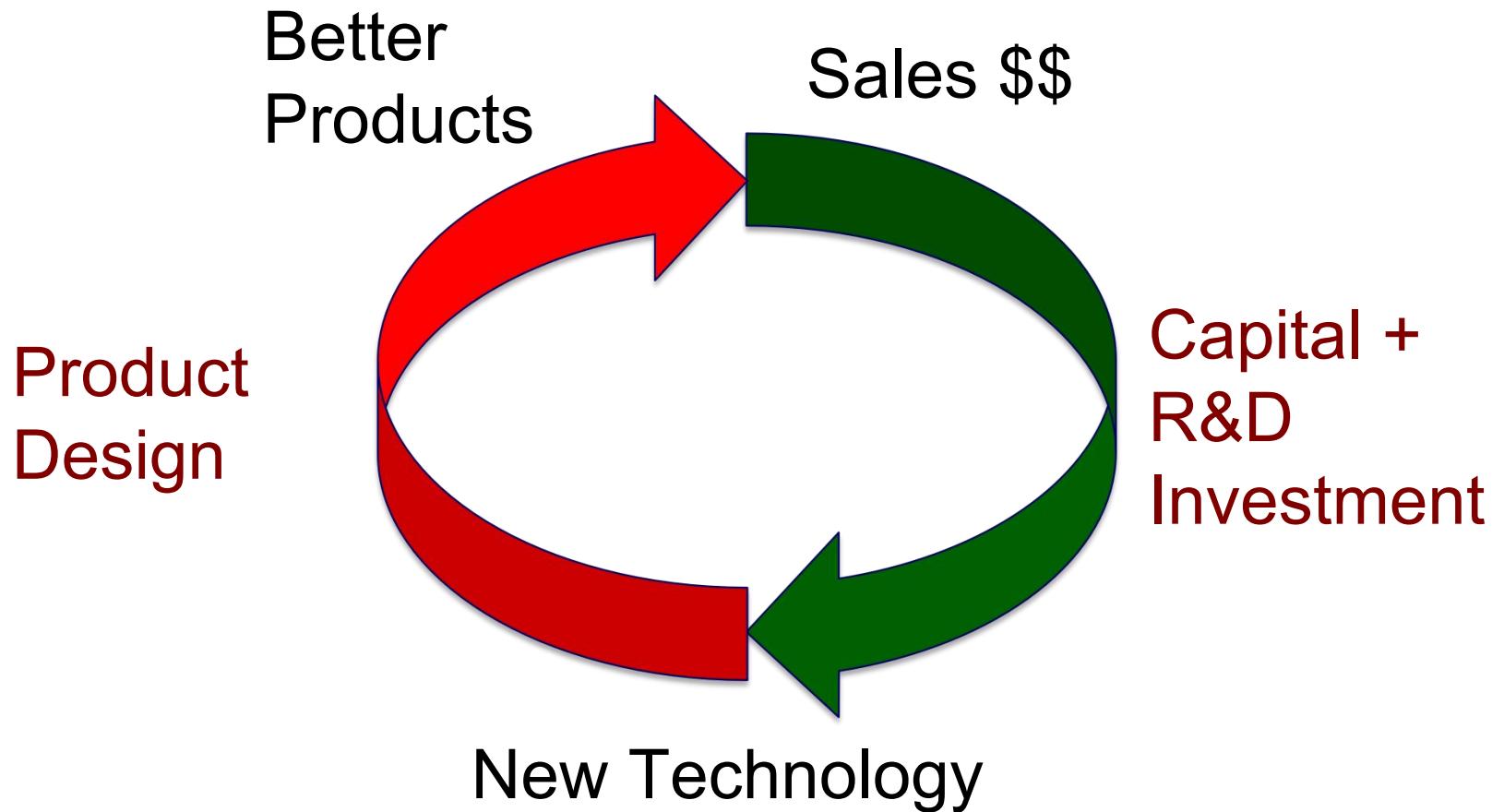
0.1 g

HiSilicon S9000
2020
15.3B transistors



~360 kg

Moore's Law Economics



**Consumer products sustain the
\$300B semiconductor industry**

What Moore's Law Has Meant

22 generations of iPhone since 2007

iPhone: June 29, 2007

iPhone 3G: July 11, 2008

iPhone 3GS: June 19, 2009

iPhone 4: June 24, 2010

iPhone 4S: October 14, 2011

iPhone 5: September 21, 2012

iPhone 5S & 5C: September 20, 2013

iPhone 6 & 6 Plus: September 19, 2014

iPhone 6S & 6S Plus: September 19, 2015

iPhone 8 & 8 Plus: September 22, 2017

iPhone X: November 3, 2017

iPhone XS, XS Max: September 21, 2018

iPhone XR: October 26, 2018

iPhone 11, Pro, Pro Max: September 20, 2019

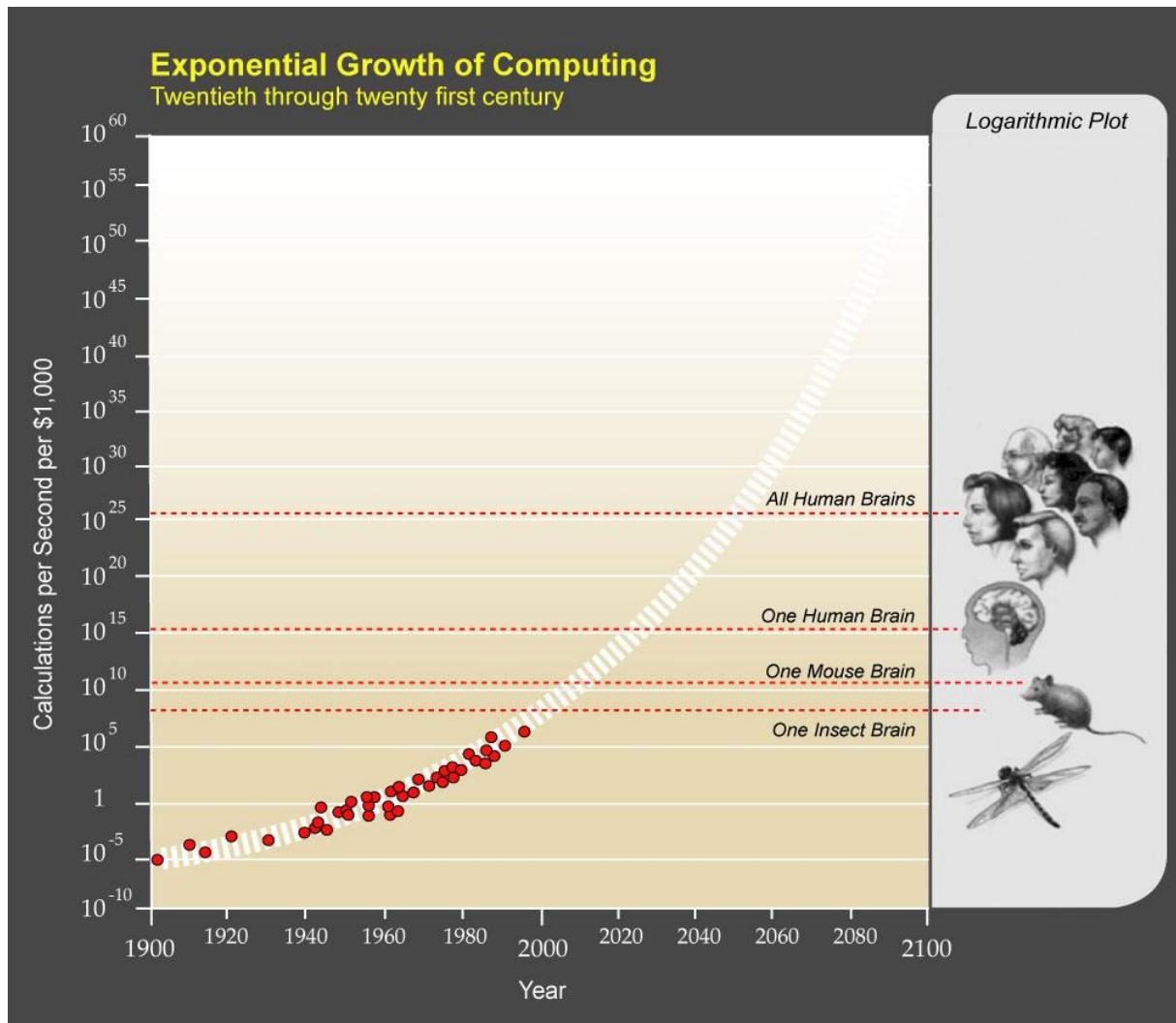
iPhone 12, Mini, Pro, Pro Max: November 13, 2020

iPhone 13, Mini, Pro, Pro Max: September 24, 2021

iPhone SE 3 (2022): March 18, 2022



What Moore's Law Could Mean



Kurzweil, *The Singularity is Near*, 2005

What Moore's Law Could Mean

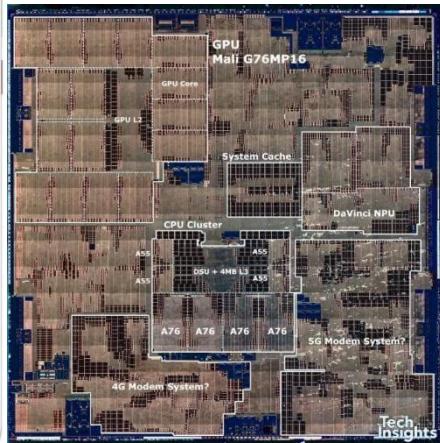
- 202x Consumer Product



- 2065 Consumer Product



- Portable
- Low power
- Will drive markets & innovation



HiSilicon Kirin 9000 with
15.3B transistors

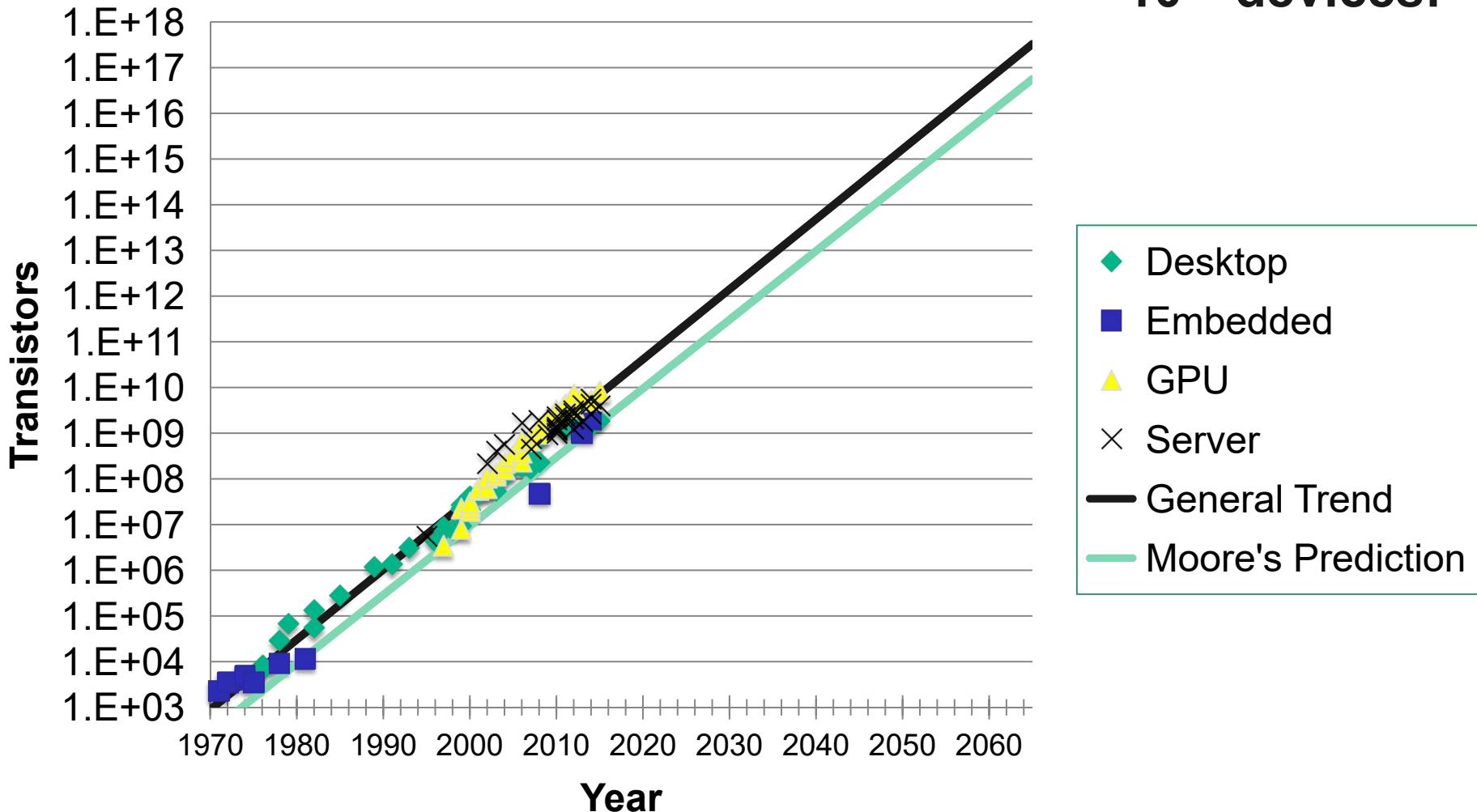
Requirements for Future Technology

- **Must be suitable for portable, low-power operation**
 - Consumer products
 - Internet of Things components
 - Not cryogenic, not quantum
- **Must be inexpensive to manufacture**
 - Comparable to current semiconductor technology
 - $O(1)$ cost to make chip with $O(N)$ devices
- **Need not be based on transistors**
 - Memristors, carbon nanotubes, DNA transcription, ...
 - Possibly new models of computation
 - But, still want lots of devices in an integrated system

Moore's Law: 100 Years

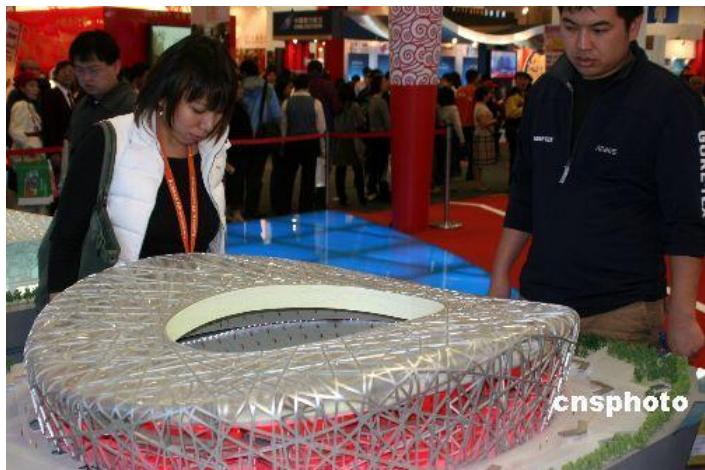
Device Count by Year

10^{17} devices!



Visualizing 10^{17} Devices

*If devices were the
size of a grain of sand*



0.1 m^3
 3.5×10^9 grains



+



1 million m^3
 0.35×10^{17} grains

Increasing Transistor Counts

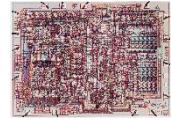
- 1. Chips have gotten bigger**
 - 1 area doubling / 10 years
- 2. Transistors have gotten smaller**
 - 4 density doublings / 10 years

Will these trends continue?

Chips Have Gotten Bigger

Intel 4004

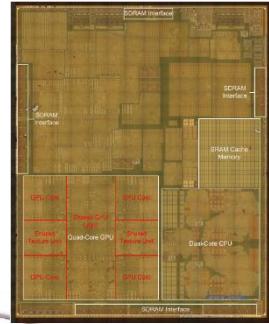
1971



Apple A8

2014

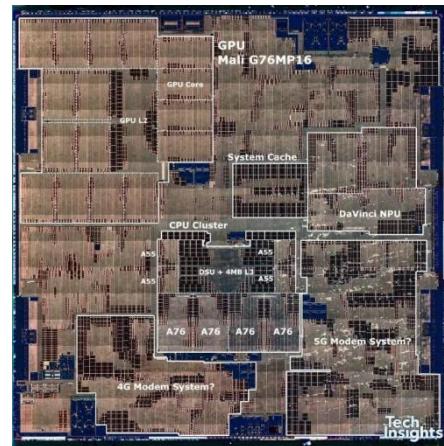
89 mm²



HiSilicon S9000

2020

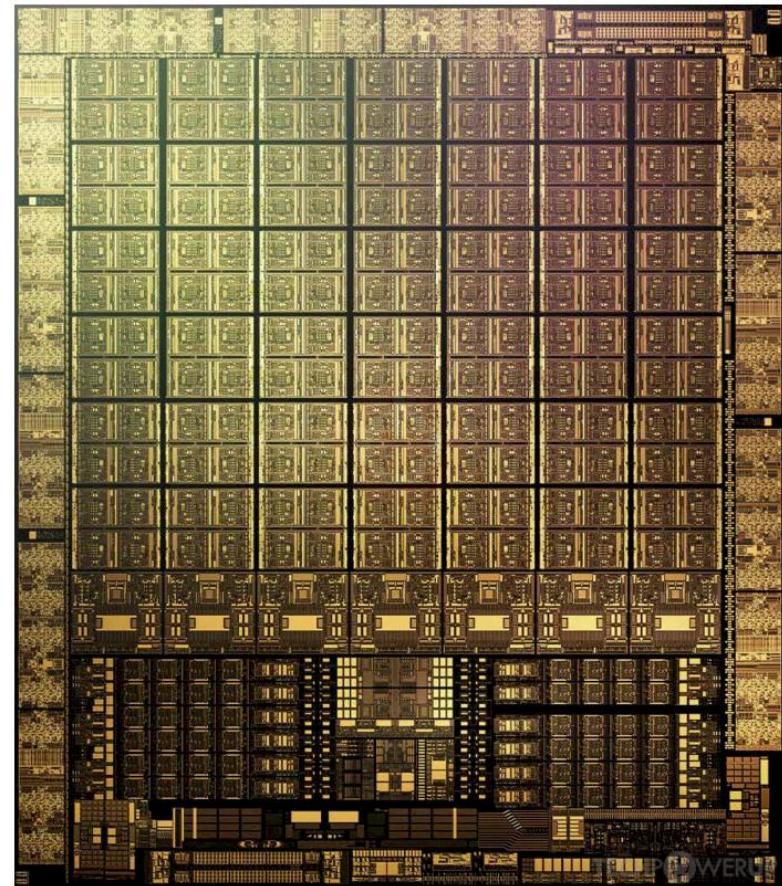
110 mm²



NVIDIA A100

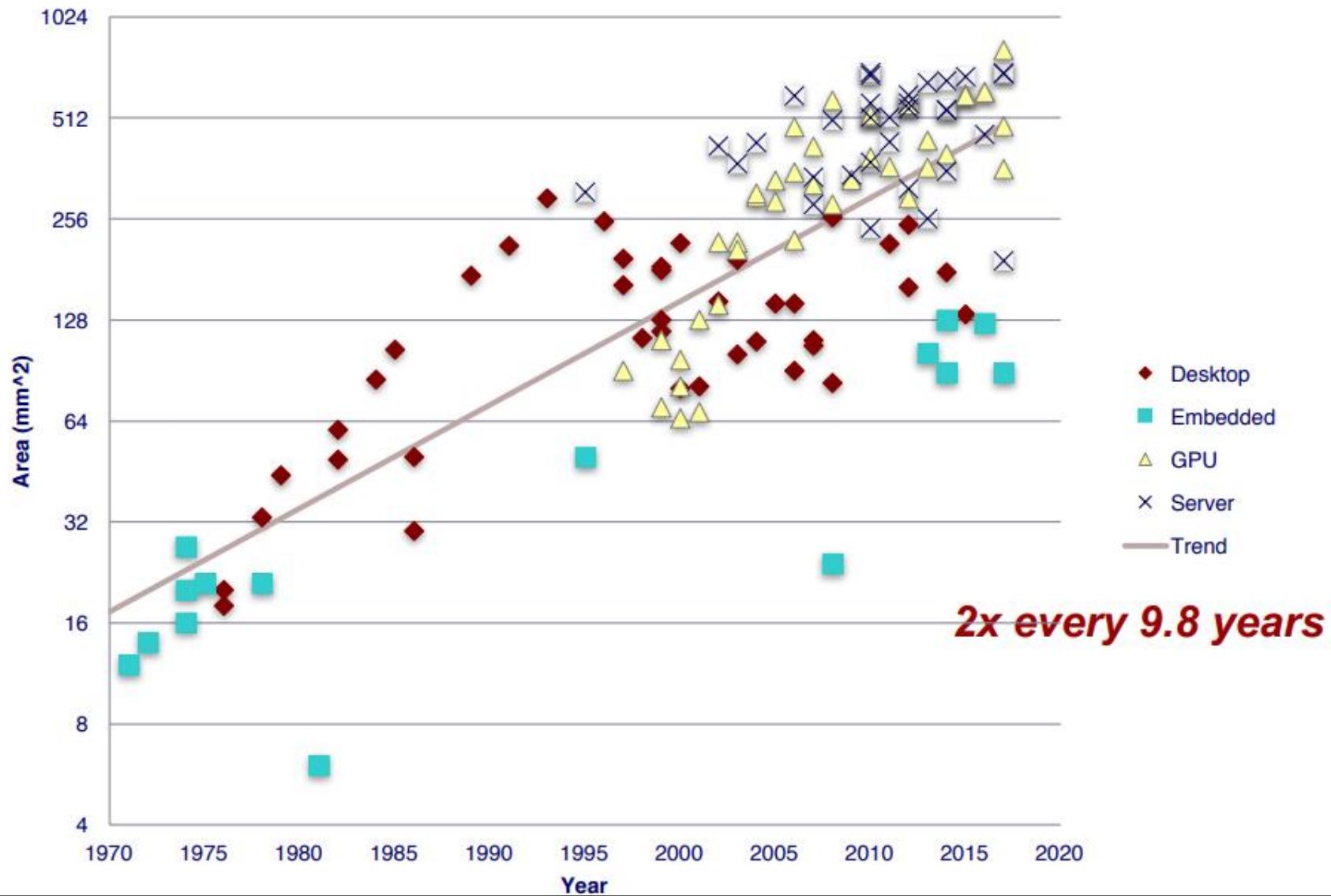
2020

826 mm²

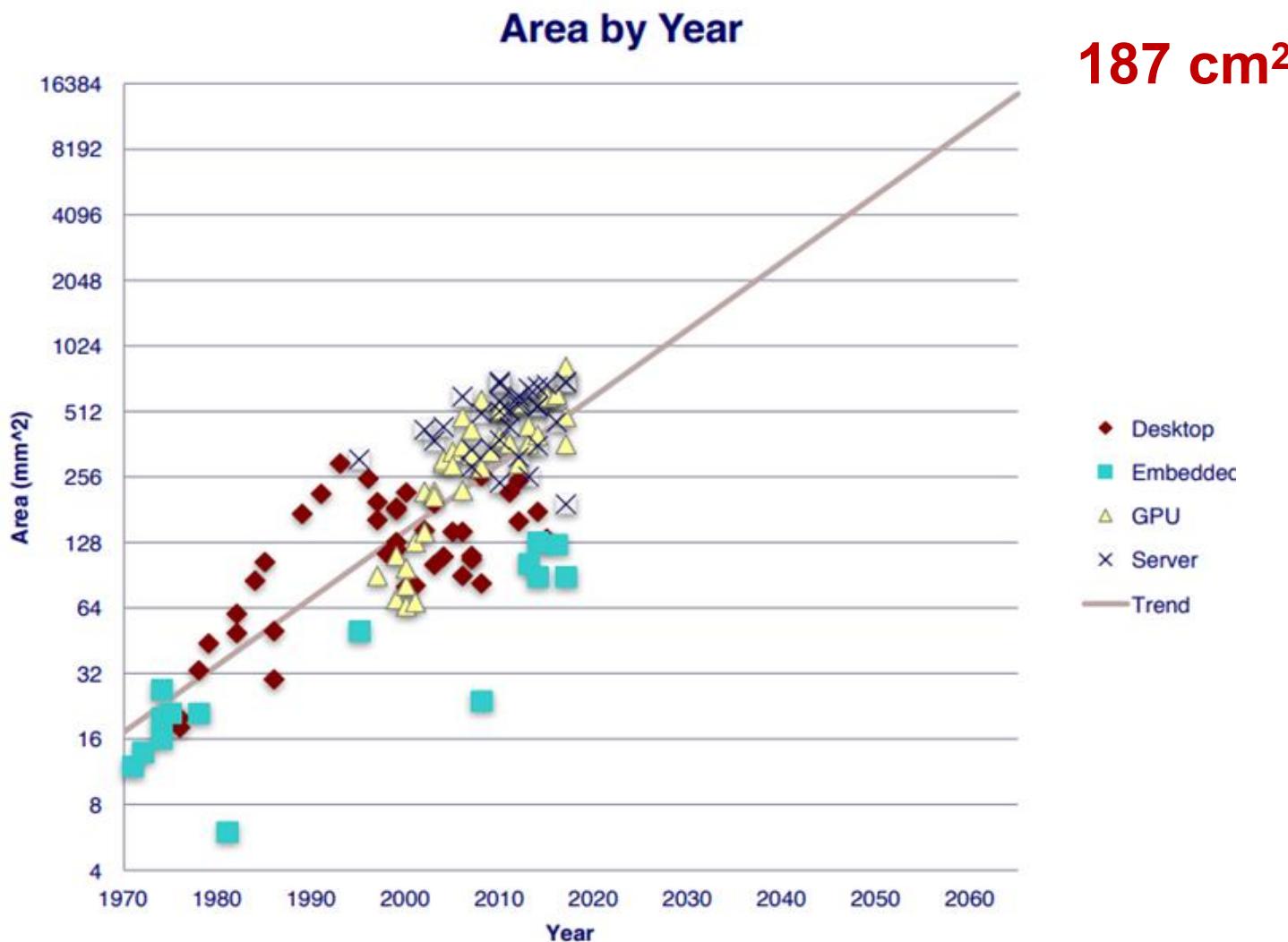


Chip Size Trend

Area by Year

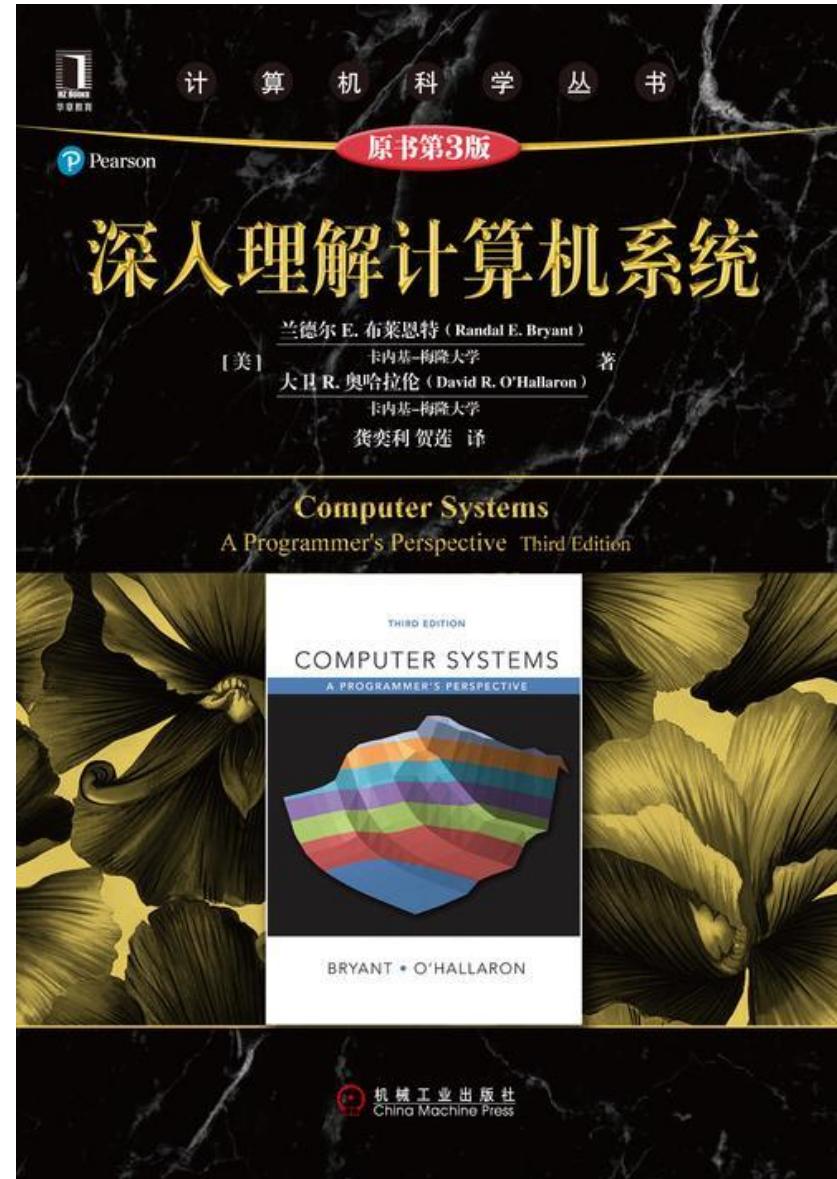


Chip Size Extrapolation



Extrapolation: The iPhone 101s

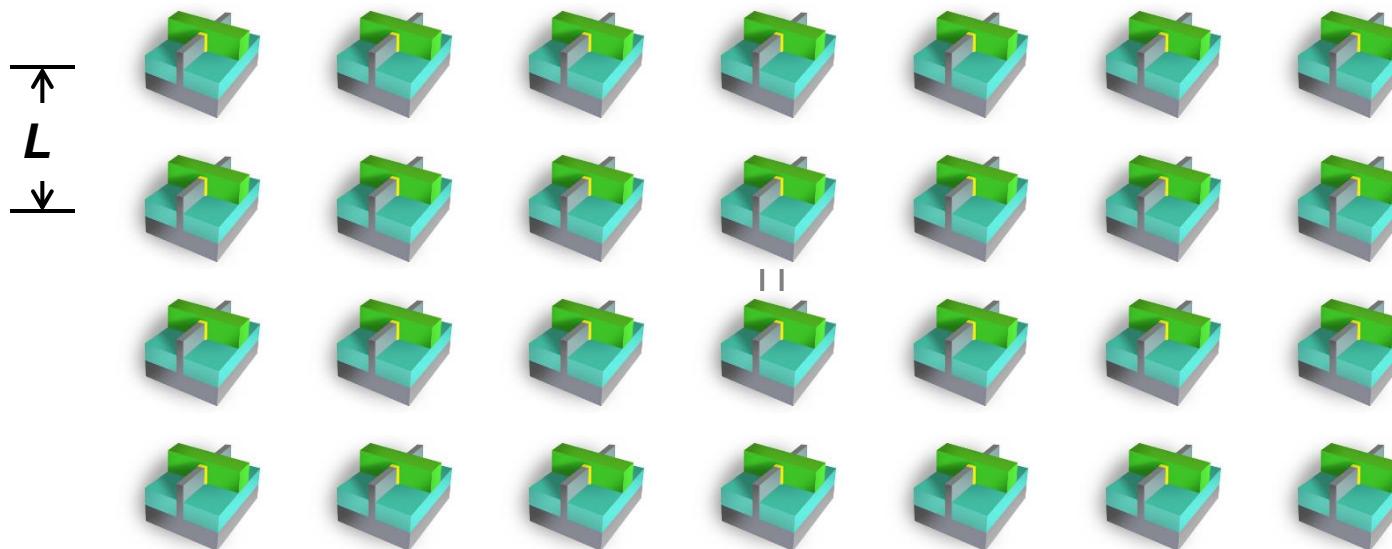
Apple Axxxx
2065
 10^{17} transistors
 187 cm^2



Transistors Have Gotten Smaller

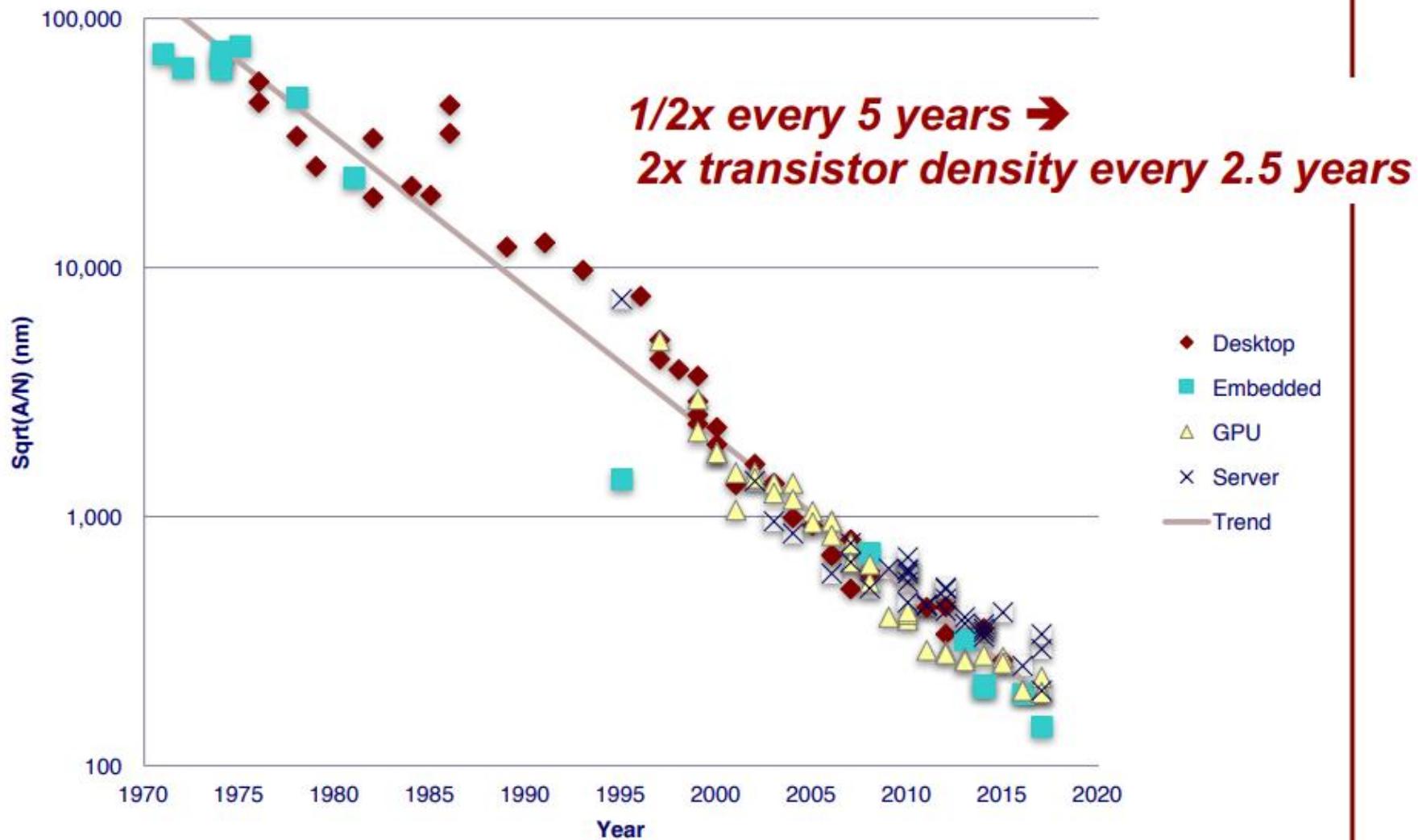
- Area A
- N devices
- Linear Scale L

$$L = \sqrt{A / N}$$



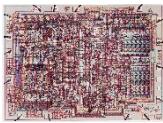
Linear Scaling Trend

Linear Spacing by Year



Decreasing Feature Sizes

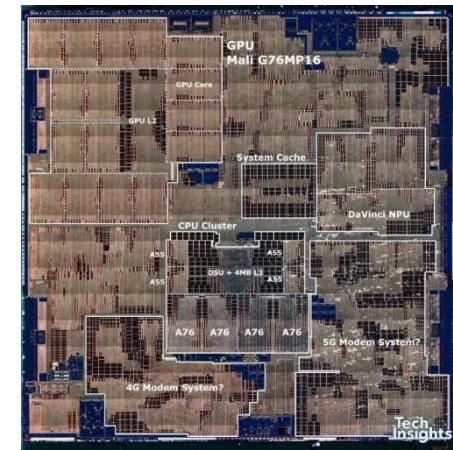
Intel 4004
1971
2,300 transistors
 $L = 72,000 \text{ nm}$



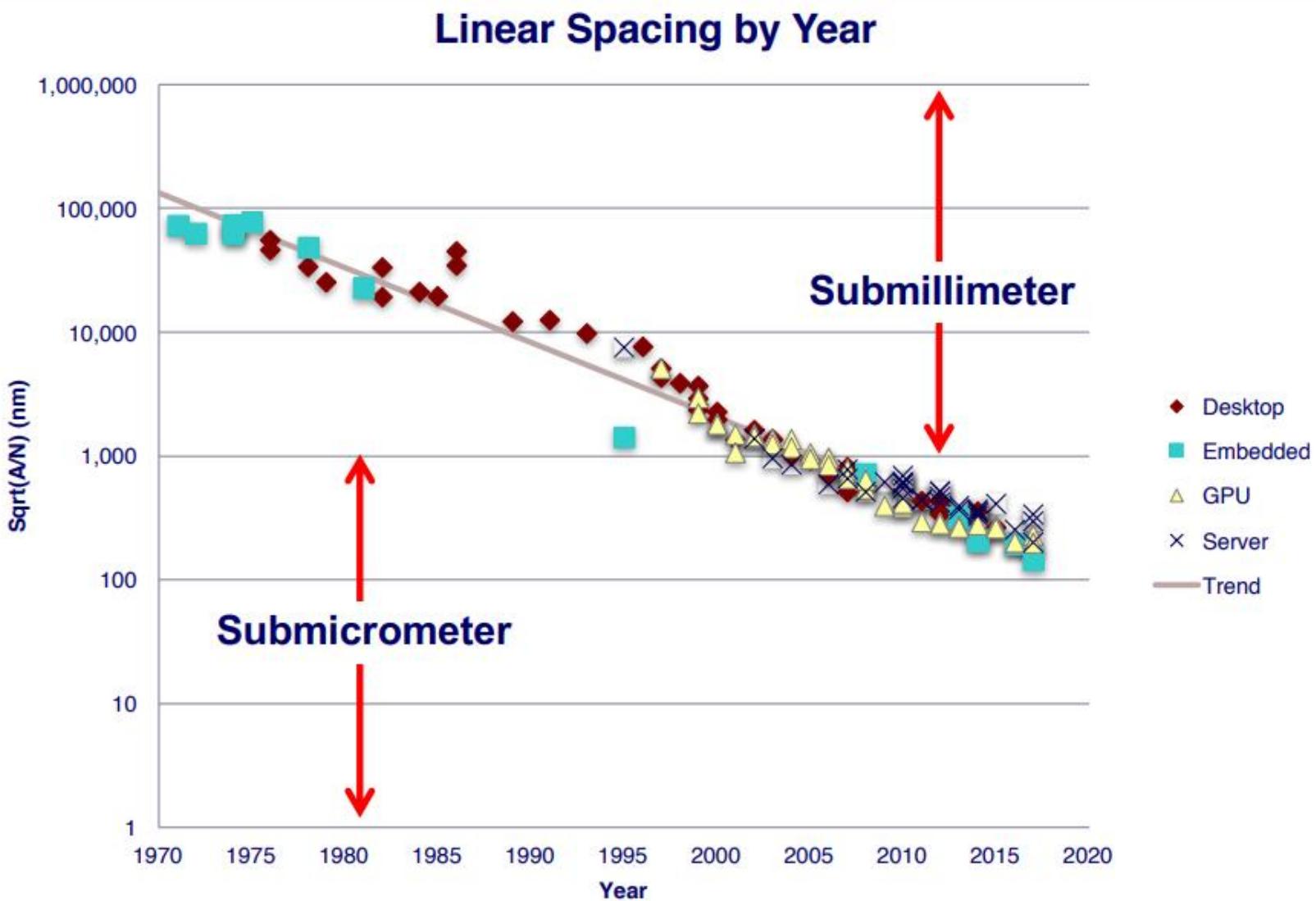
Apple A8
2014
2 B transistors
 $L = 211 \text{ nm}$



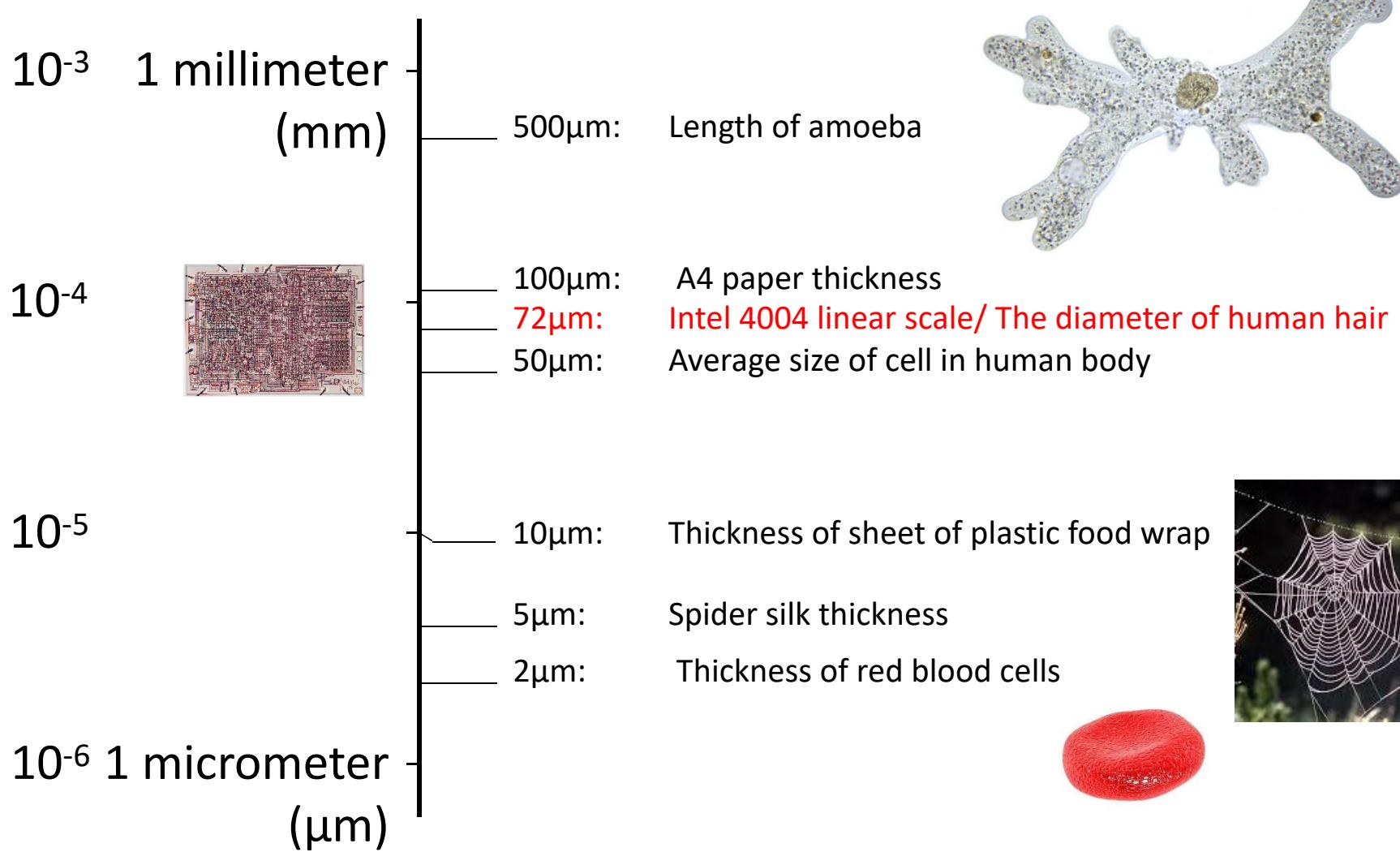
HiSilicon S9000
2020
15.3B transistors
 $L= 85\text{nm}$



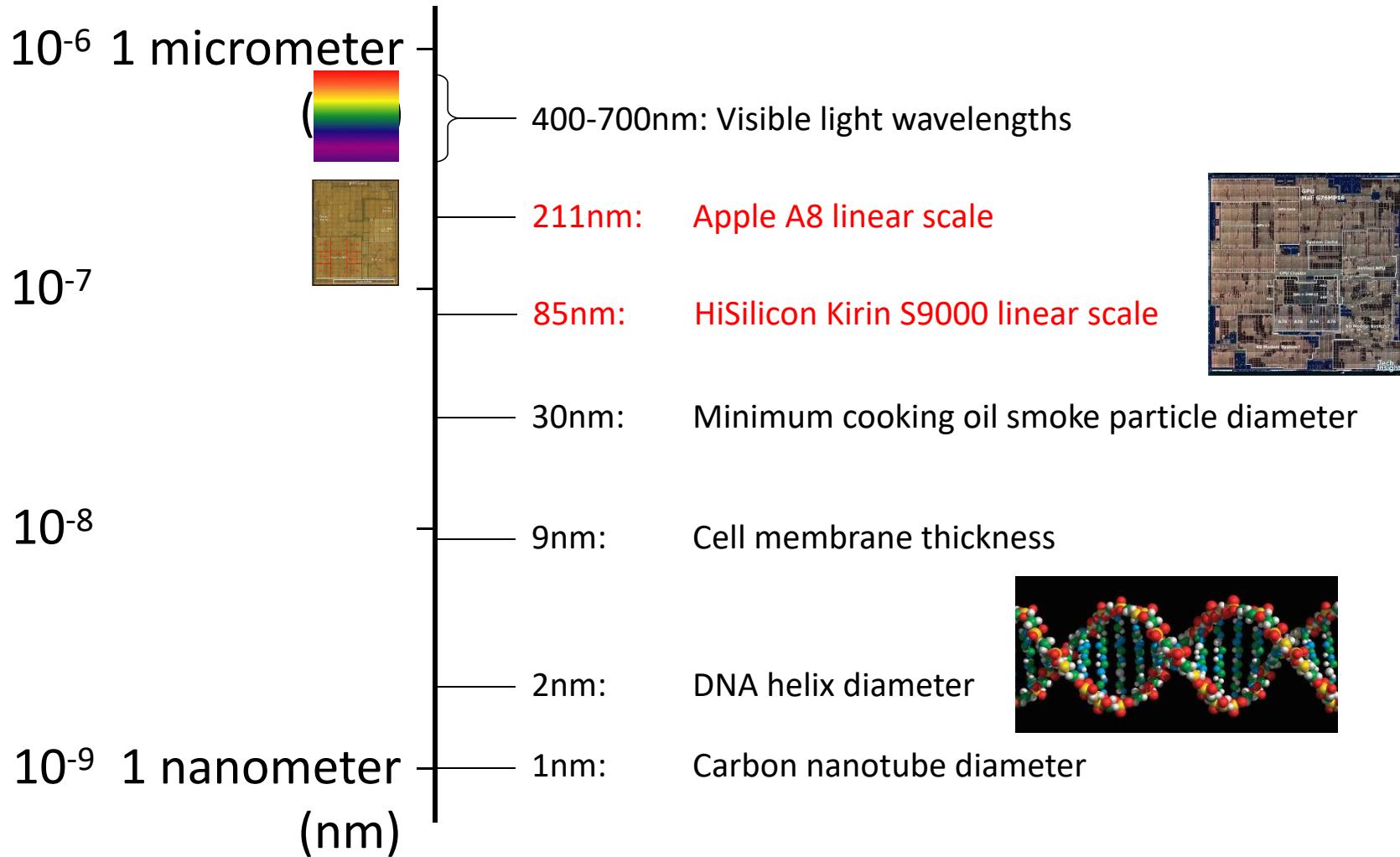
Linear Scaling Trend



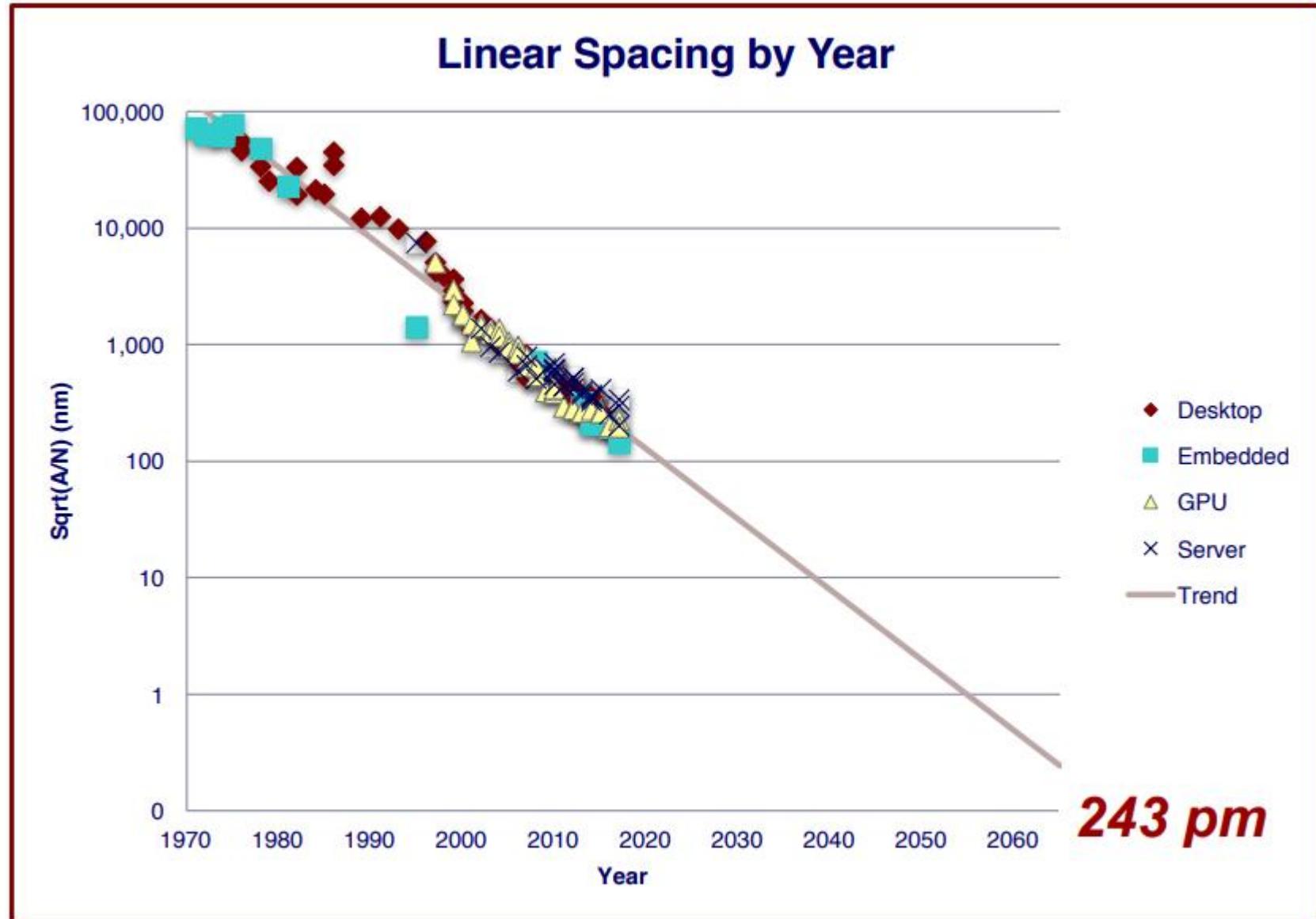
Submillimeter Dimensions



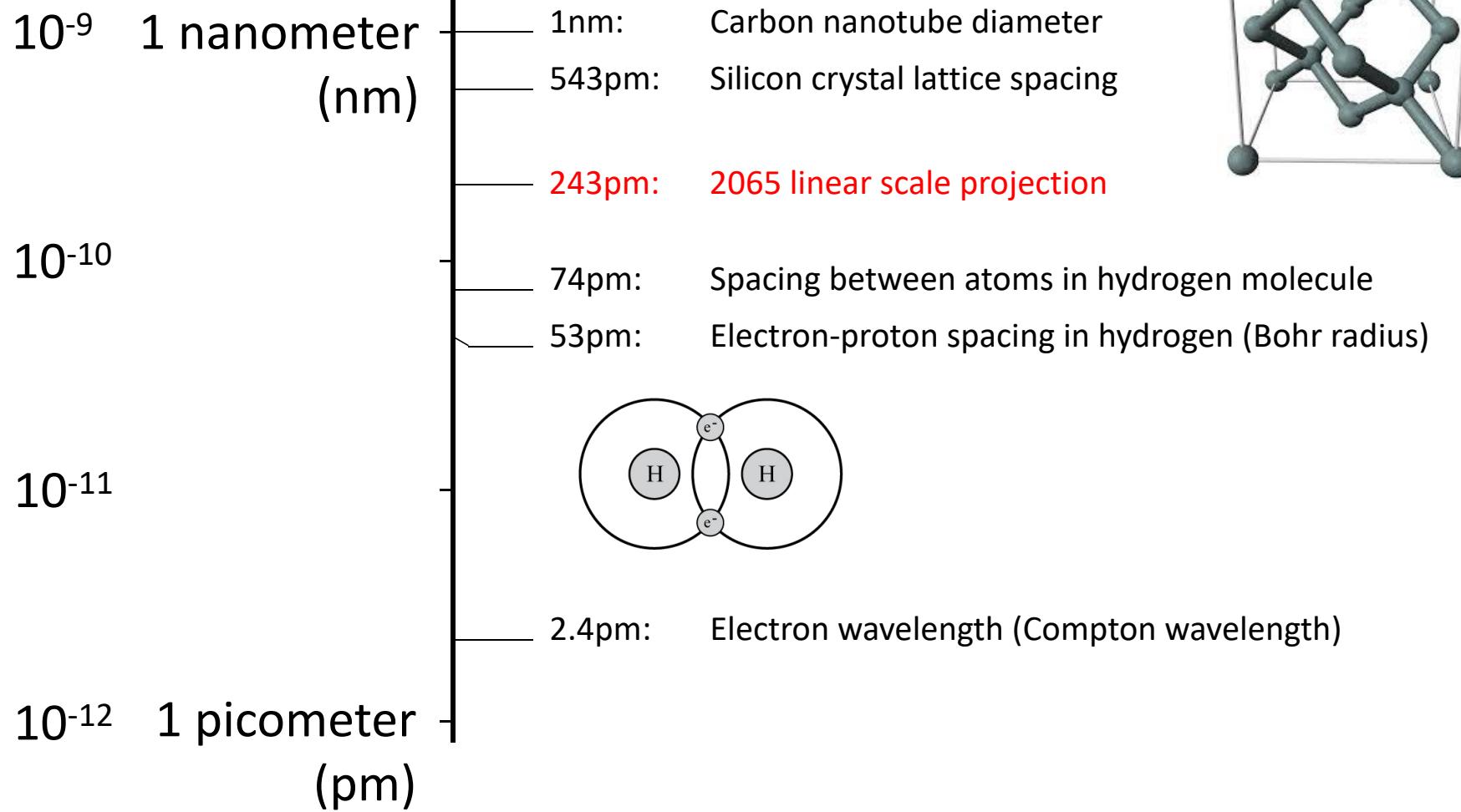
Submicrometer Dimensions



Linear Scaling Extrapolation



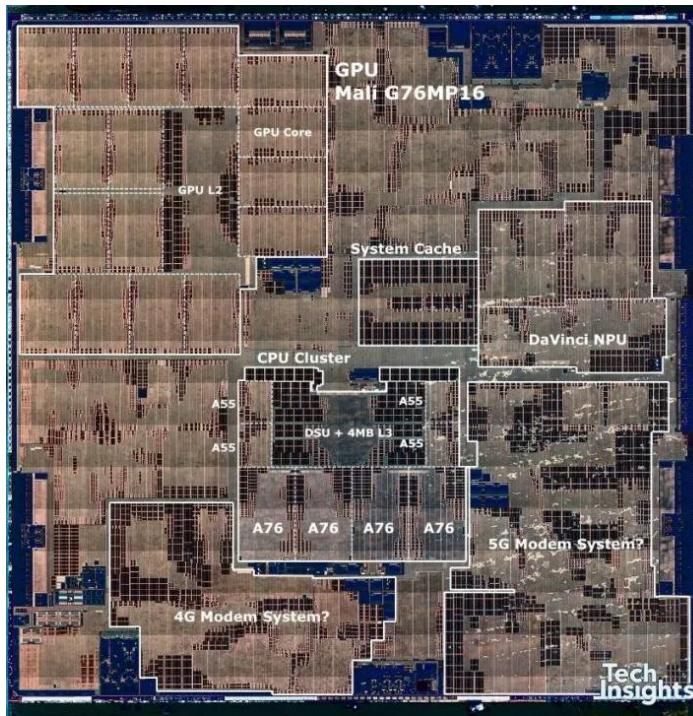
Subnanometer Dimensions



Reaching 2065 Goal

- Target

- 10^{17} devices
- 400 mm²
- $L = 63$ pm

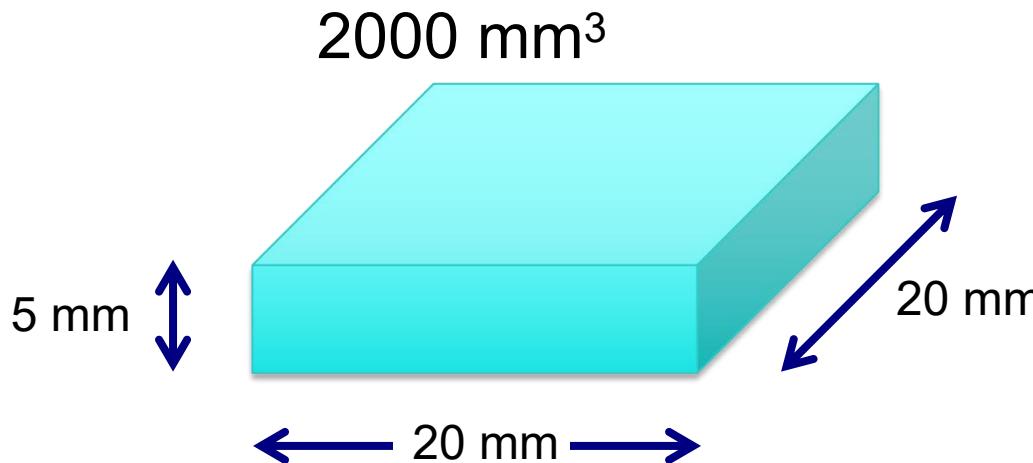


- Is this possible?

No!

Not with 2-D
fabrication

Fabricating in 3 Dimensions



○ Parameters

- 10^{17} devices
- 100,000 logical layers
 - Each 50 nm thick
 - $\sim 1,000,000$ physical layers
 - To provide wiring and isolation
- $L = 20 \text{ nm}$
 - 10x smaller than today



$\approx 2000 \text{ mm}^3$

3D Fabrication Challenges

- **Yield**

- How to avoid or tolerate flaws

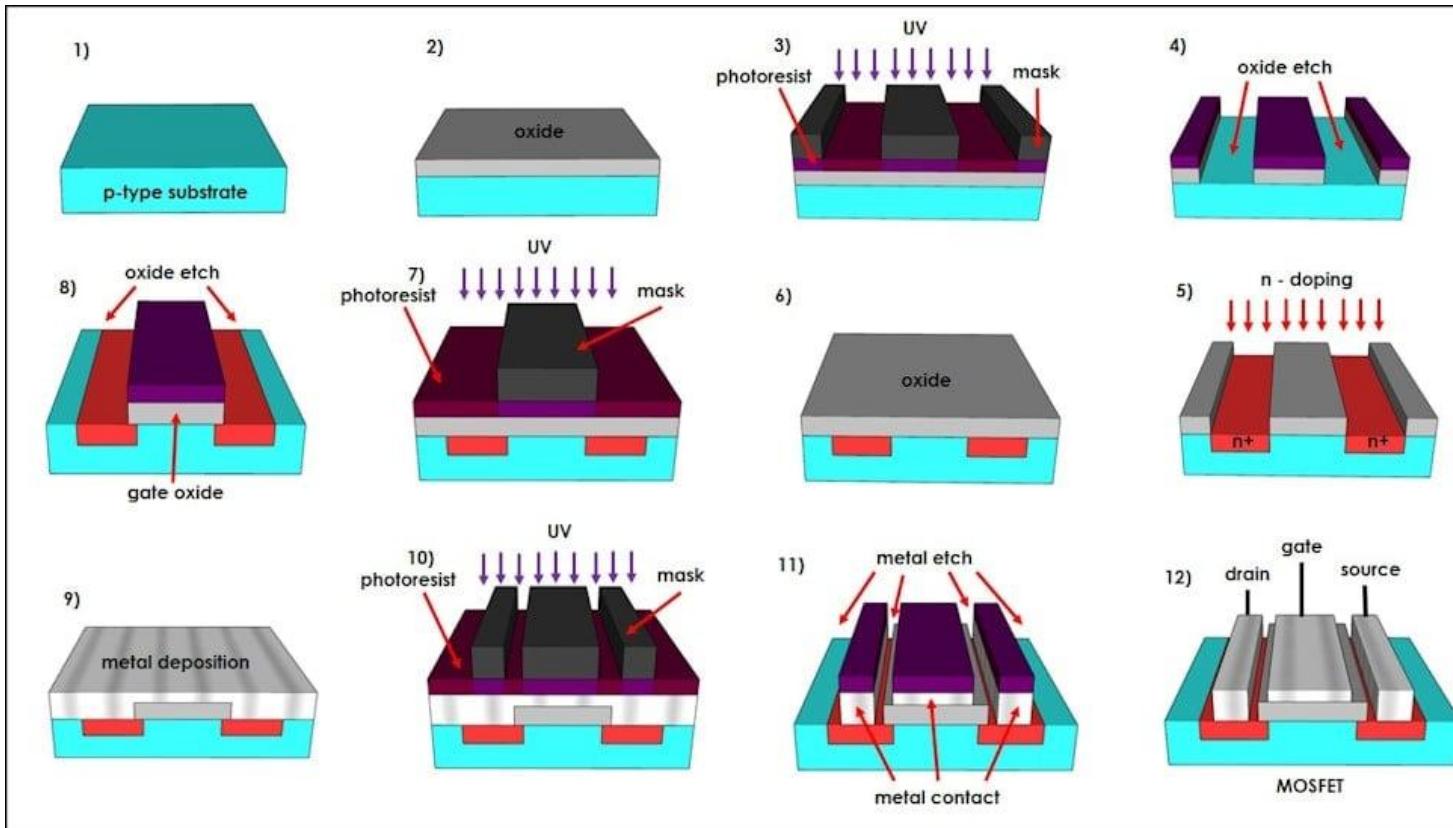
- **Cost**

- High cost of lithography

- **Power**

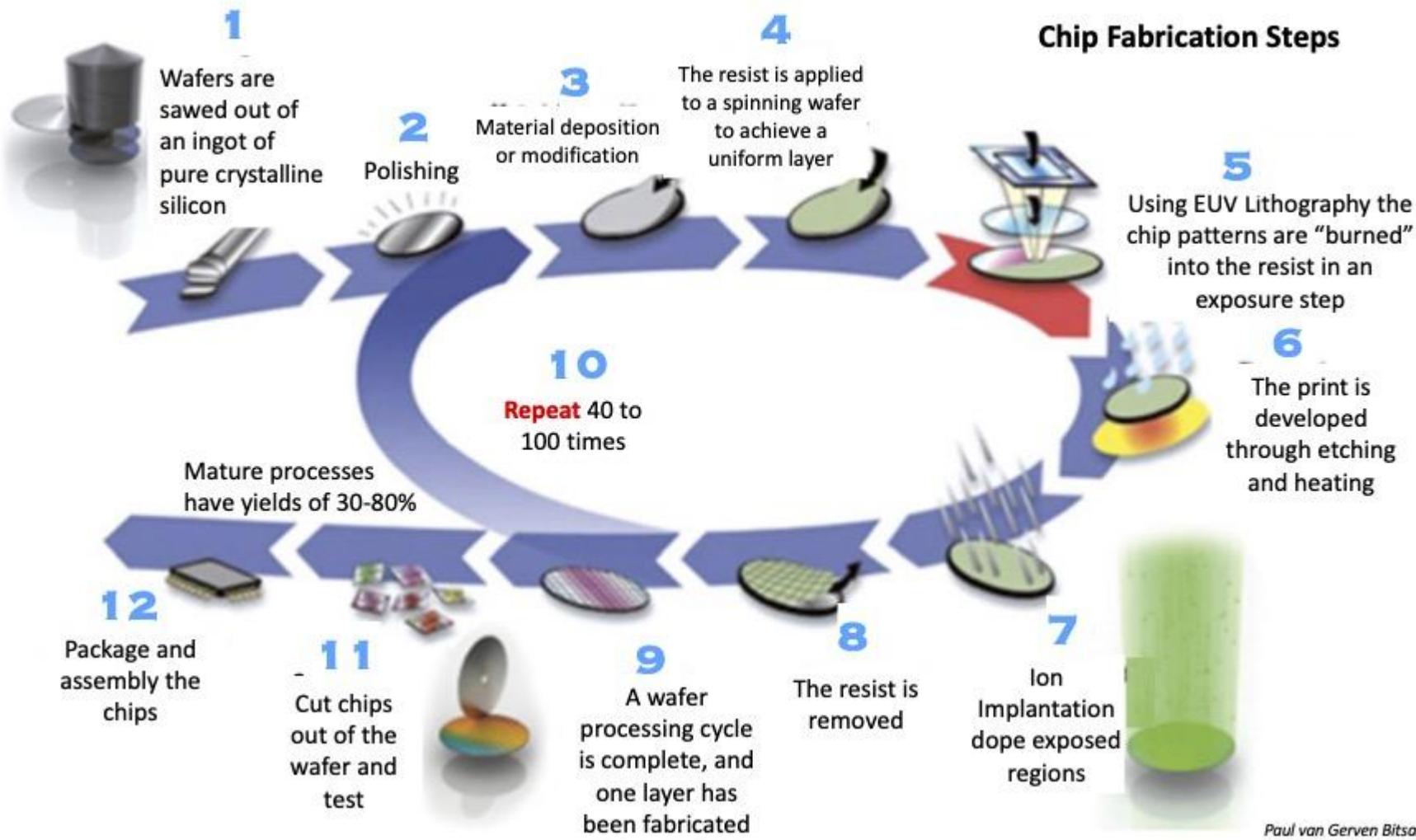
- Keep power consumption within acceptable limits
 - Limited energy available
 - Limited ability to dissipate heat

Photolithography

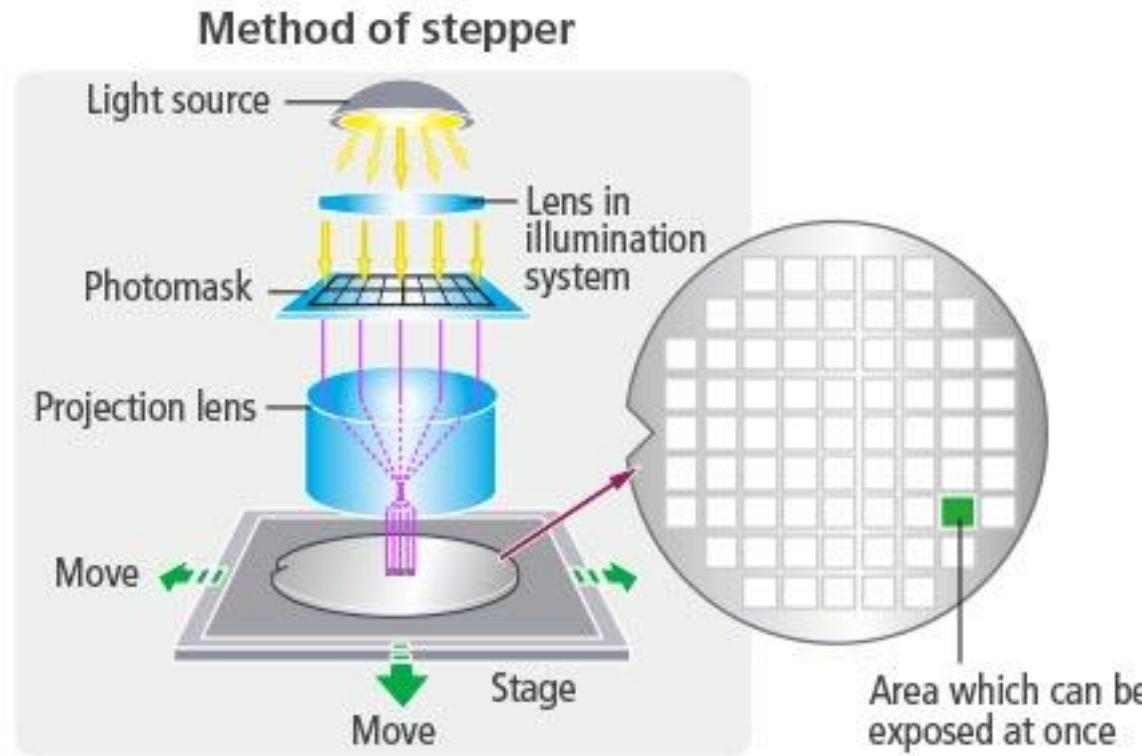


- Pattern entire chip in one step
- Modern chips require ~60 lithography steps
- Fabricate N transistor system with $O(1)$ steps

Semiconductor Fabrication Process



Fabrication Costs



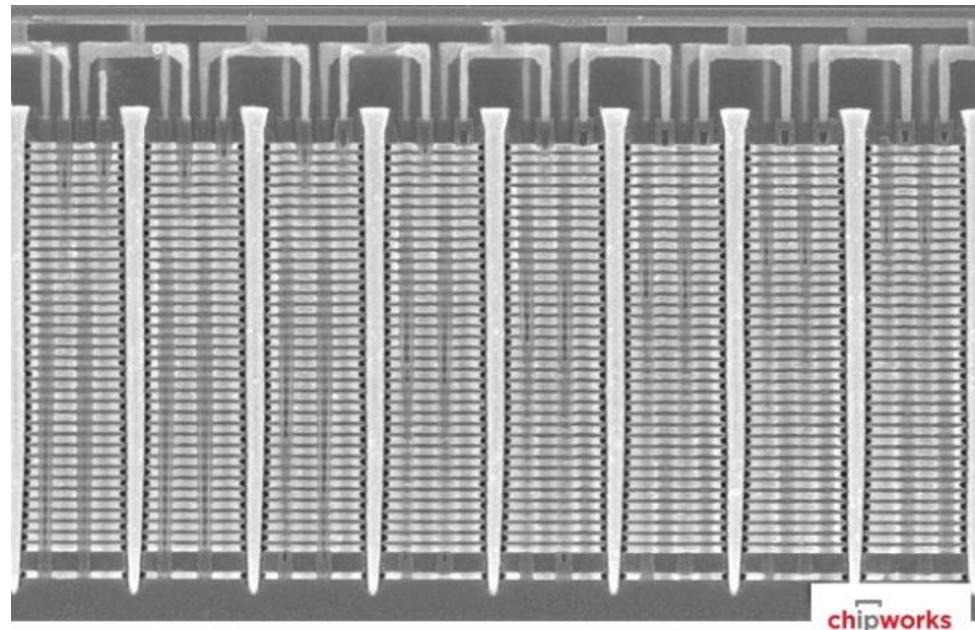
○ Stepper

- Most expensive equipment in fabrication facility
- Rate limiting process step
 - 18s / wafer
- Expose 858 mm^2 per step
 - 1.2% of chip area

Fabrication Economics

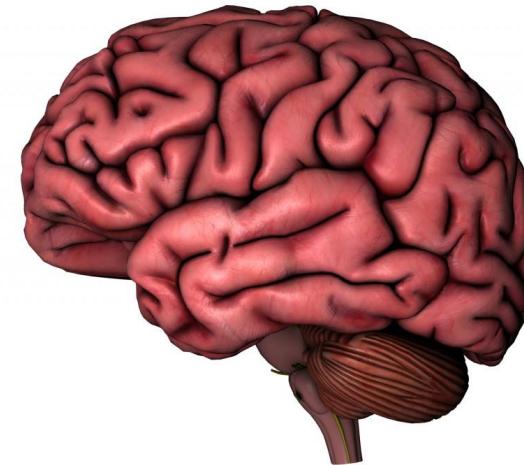
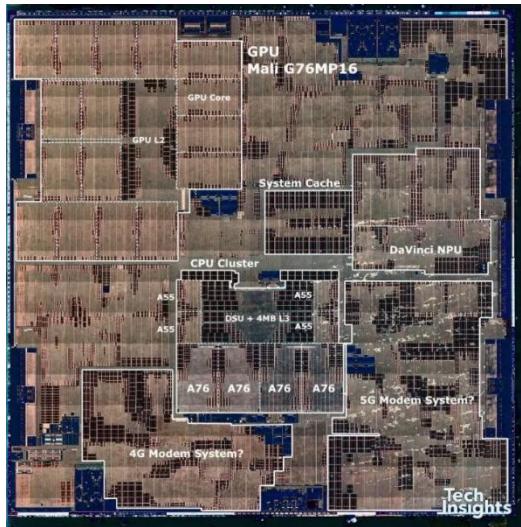
- **Currently**
 - Fixed number of lithography steps
 - Manufacturing cost \$10–\$20 / chip
 - Including amortization of facility
- **Fabricating 1,000,000 physical layers**
 - Cannot do lithography on every step
- **Options**
 - Chemical self assembly
 - Devices generate themselves via chemical processes
 - Pattern multiple layers at once

Samsung V-Nand Flash Example



- Build up layers of unpatterned material
- Then use lithography to slice, drill, etch, and deposit material across all layers
- Over 30 total masking steps
- 300+ layers of memory cells (SK Hynix, 2023.08)
- Exploits particular structure of flash memory circuits

Meeting Power Constraints



- 15.3 B transistors
 - 2.62 GHz operation
 - 1–5 W
 - 64 B neurons
 - 100 Hz operation
 - 15–25 W
 - Liquid cooling
 - Up to 25% body's total energy consumption
- Can we increase number of devices by 500,000x without increasing power requirement?***

Challenges to Moore's Law: Economic

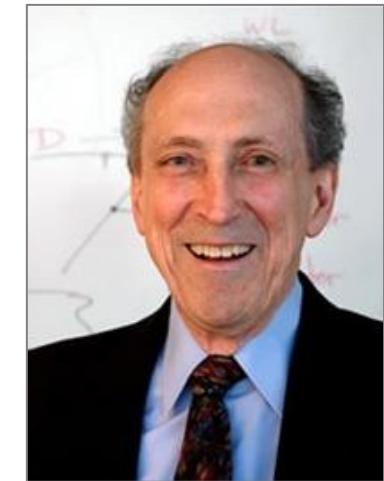
- Growing Capital Costs

- State of art fab line > \$20B
- Must have very high volumes to amortize investment
- Has led to major consolidations



Dennard Scaling

Due to Robert Dennard, IBM, 1974
 Quantifies benefits of Moore's Law



○ How to shrink an IC Process

- Reduce horizontal and vertical dimensions by k
- Reduce voltage by k

○ Outcomes

- Devices / chip increase by k^2
- Clock frequency increases by k
- Power / chip constant

○ Significance

- Increased capacity and performance
- No increase in power

Design of Ion-Implanted MOSFET's with Very Small Physical Dimensions

ROBERT H. DENNARD, MEMBER, IEEE, FRITZ H. GAENSSLER, HWA-NIEN YU, MEMBER, IEEE,
 V. LEO RIDEOUT, MEMBER, IEEE, ERNEST BASSOUS, AND ANDRE R. LEBLANC, MEMBER, IEEE

Classic Paper

q	Charge on the electron.
Q_{it}	Effective oxide charge.
T_{th}	Gate threshold voltage.
T	Absolute temperature.
V_D , V_G , V_S , V_{th}	Drain, source and substrate voltages.
V_{ds}	Drain-to-source voltage relative to source.
V_{dsat}	Source voltage relative to substrate.
V_{th}	Gate threshold voltage.
w , L	Source and drain depletion layer width.
W	MOSFET channel width.

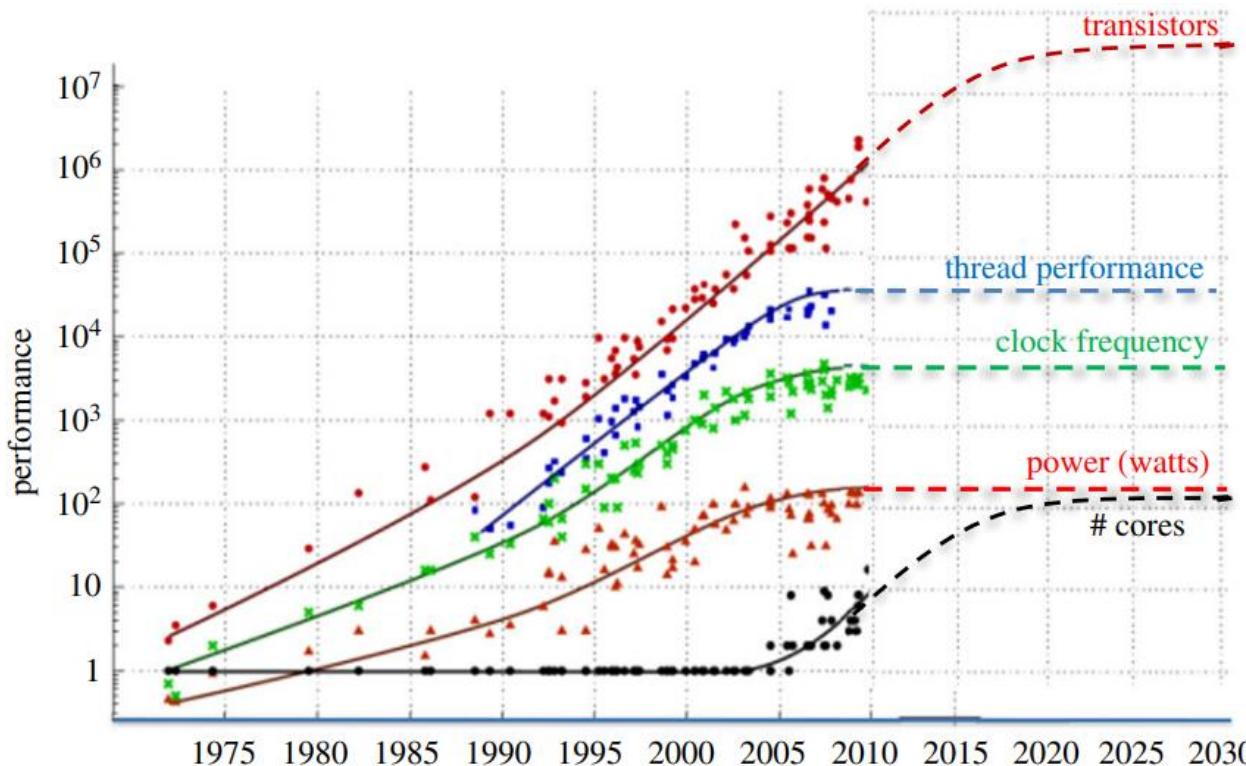
II. INTRODUCTION

New high resolution lithographic techniques for forming semiconductor integrated circuit patterns offer a decrease in linewidths of five to ten times over the optical contact masking approach which is currently in widespread use. One of these tools, Of course, electron beam patterning, has been widely used for experimental device fabrication [1]-[4] while X-ray lithography [5] and optical projection printing [6] have also exhibited high resolution capabilities. In view of the tremendous potential for new high resolution lithographic techniques, the development of new device designs, technologies, and structures which can be optimized for very small dimensions.

The paper concerns the design, fabrication, and characterization of very small MOSFET's switching devices suitable for digital integrated circuits using dimensions of the order of 1μ . It is known that reducing the source-to-drain spacing (i.e., the channel length) of an FET leads to noticeable changes in the drain characteristics. These changes become more pronounced when the depletion regions surrounding the source and drain extend over a large portion of the region in the silicon substrate under the gate electrode. For switching applications, the most undesirable "short-channel" effect is a reduction in the gate threshold voltage at which the device turns on, which is aggravated

This paper is reprinted from IEEE JOURNAL OF SOLID STATE CIRCUITS, vol. SC-9, no. 3, pp. 256-268, October 1974.
 Publisher from Monelle. S.0018-9290(83)02296-8.

End of Dennard Scaling



○ What Happened?

- Can't drop voltage below $\sim 1V$
- Reached limit of power / chip in 2004
- More logic on chip (Moore's Law), but can't make them run faster
 - Response has been to increase cores / chip

Some Thoughts about Technology

- **Compared to future, past 50 years will seem fairly straightforward**
 - 50 years of using photolithography to pattern transistors on two-dimensional surface
- **Questions about future integrated systems**
 - Can we build them?
 - What will be the technology?
 - Are they commercially viable?
 - Can we keep power consumption low?
 - What will we do with them?
 - How will we program / customize them?

Future of Computing

HIGH-PERFORMANCE COMPUTING

Comparing Two Large-Scale Systems

- Oakridge Titan



- Google Data Center



- Monolithic supercomputer
(4th fastest in world
@2017, rank 63th now)
- Designed for compute-intensive applications

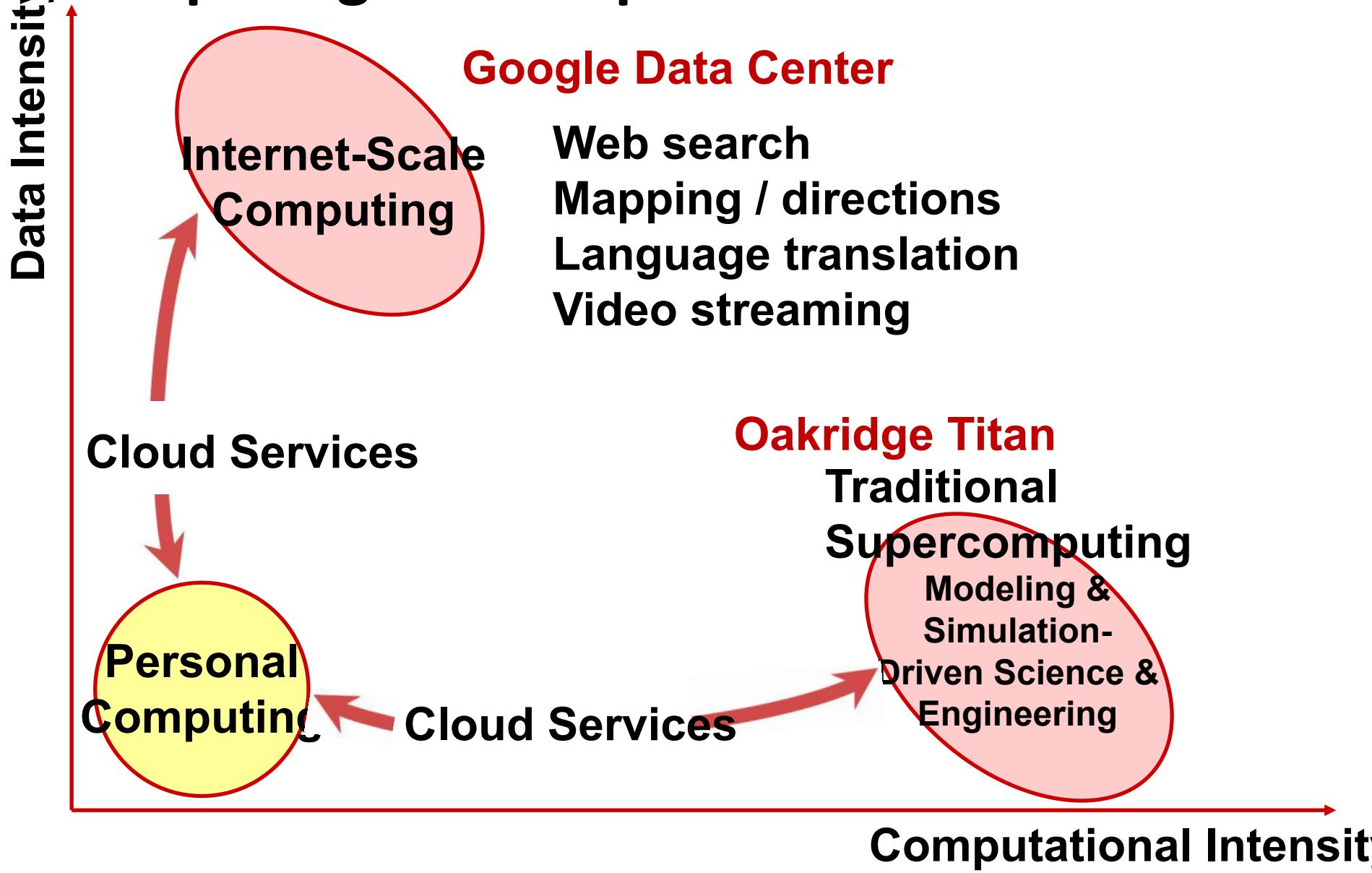
- Servers to support millions of customers
- Designed for data collection, storage, and analysis

Sunway Taihu Light

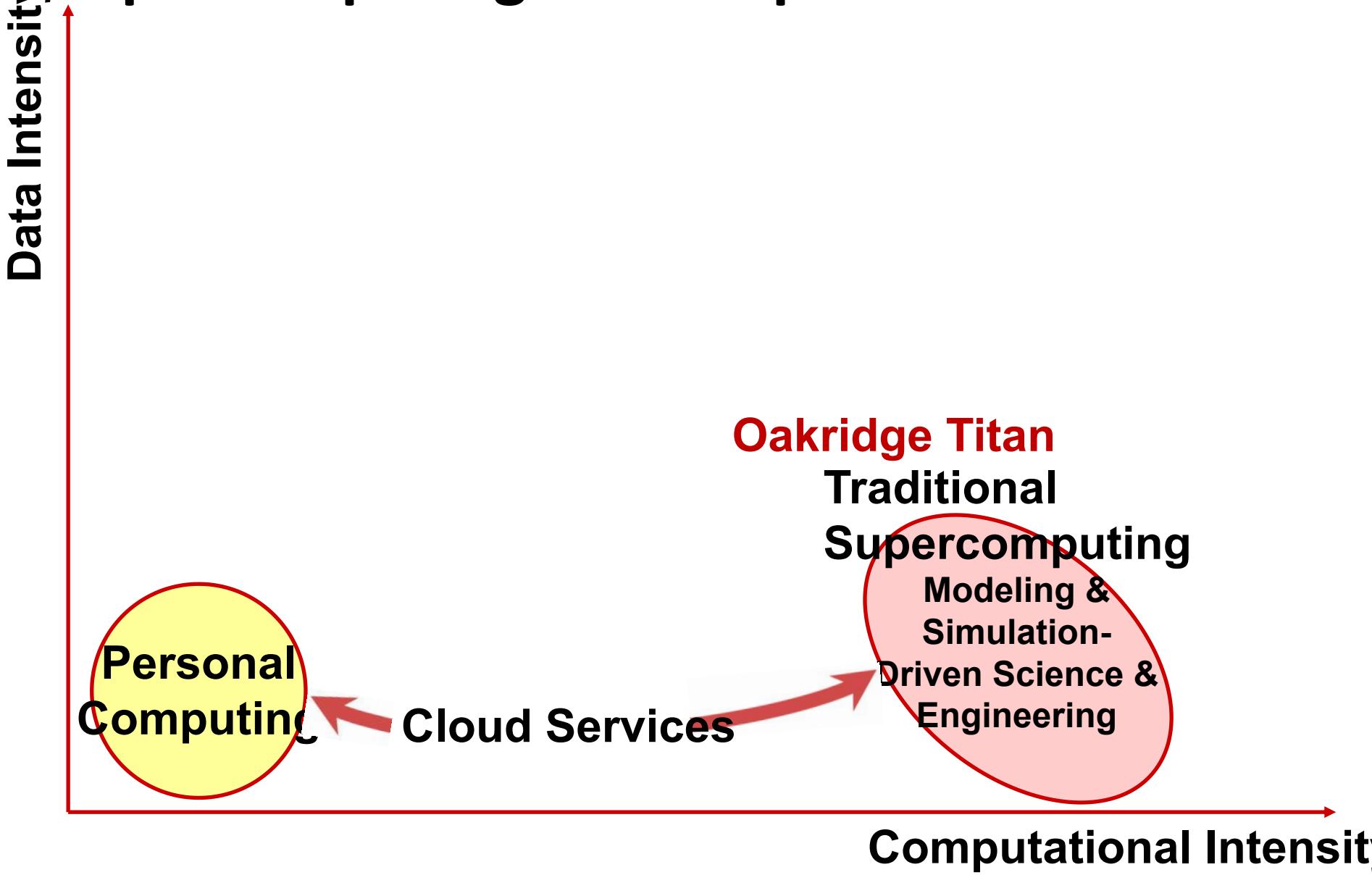
- Chinese supercomputer, ranked 15th in the TOP500 list



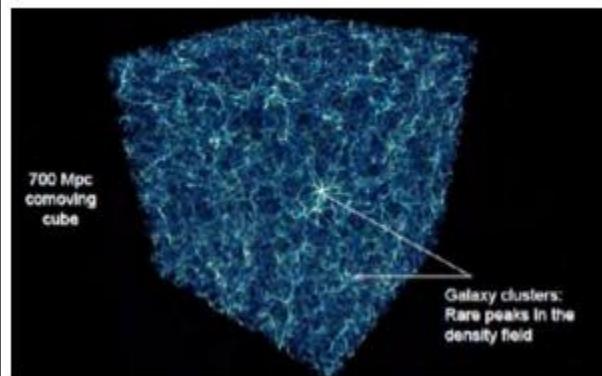
Computing Landscape



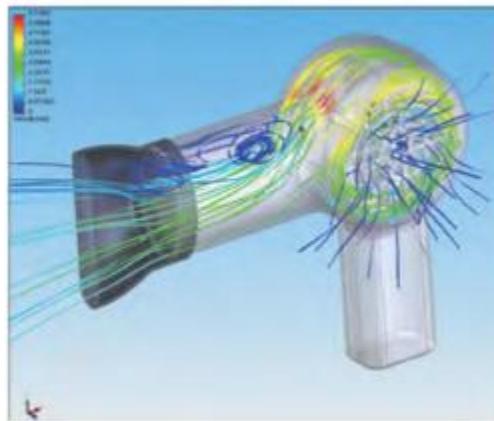
Supercomputing Landscape



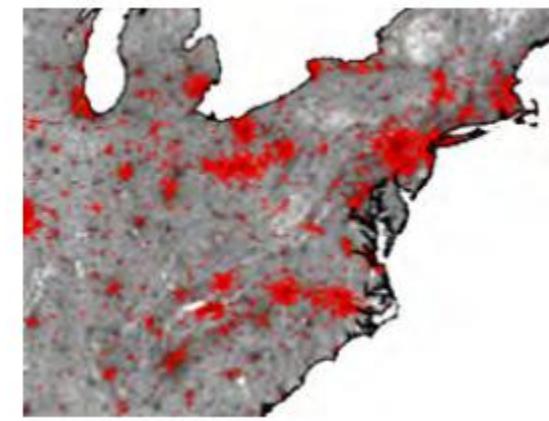
Supercomputer Applications



Science



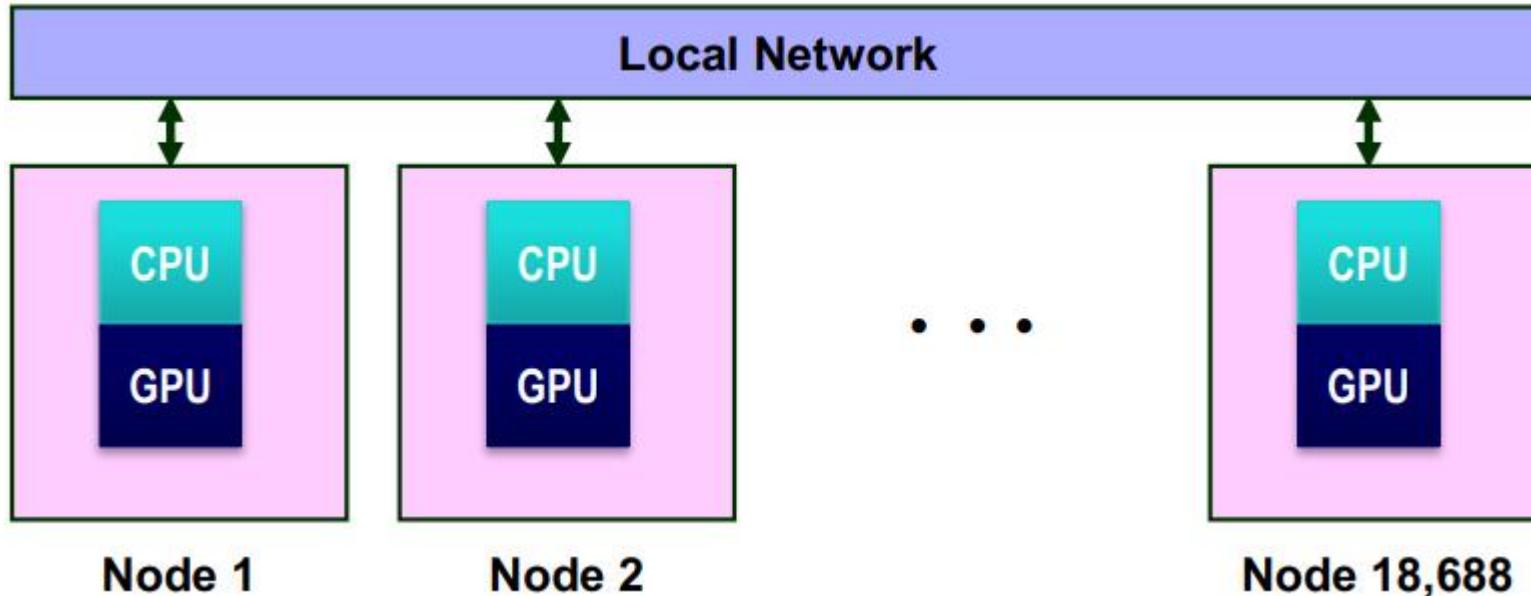
Industrial Products



Public Health

- **Simulation-Based Modeling**
 - System structure + initial conditions + transition behavior § Discretize time and space § Run simulation to see what happens
- **Requirements**
 - Model accurately reflects actual system § Simulation faithfully captures model

Titan Hardware



- **Each Node**

- AMD 16-core processor
- nVidia Graphics Processing Unit
- 38 GB DRAM
- No disk drive

- **Overall**

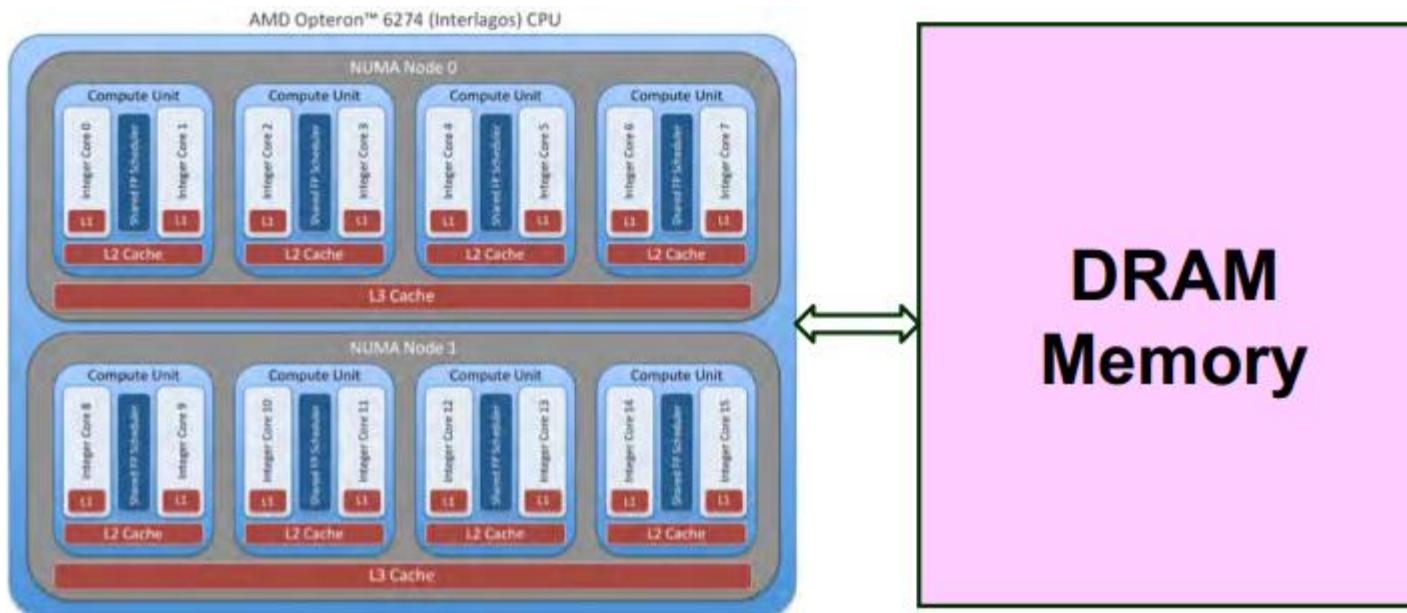
- 7MW, \$200M



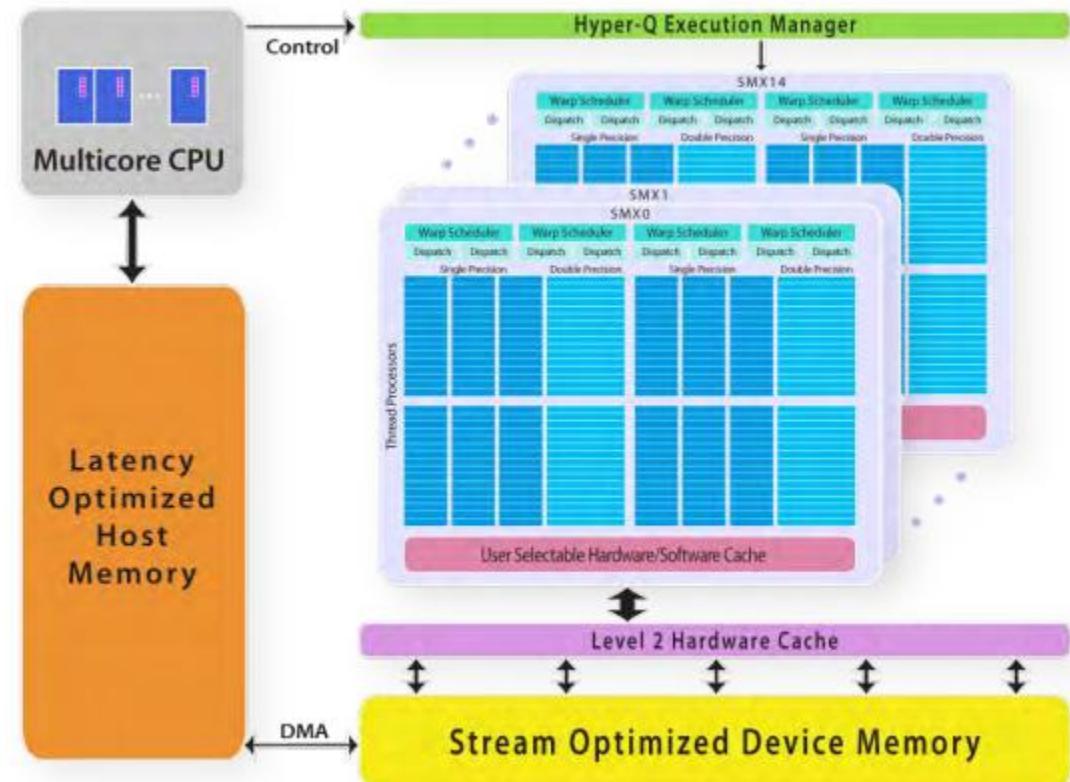
Titan Node Structure: CPU

○ CPU

- 16 cores sharing common memory
- Supports multithreaded programming
- $\sim 0.16 \times 10^{12}$ floating-point operations per second (FLOPS) peak performance



Titan Node Structure: GPU



○ Kepler GPU

- 14 multiprocessors
- Each with 12 groups of 16 stream processors
 - $14 \times 12 \times 16 = 2688$
- Single-Instruction, Multiple-Data parallelism
 - Single instruction controls all processors in group
- 4.0 x 1012 FLOPS peak performance

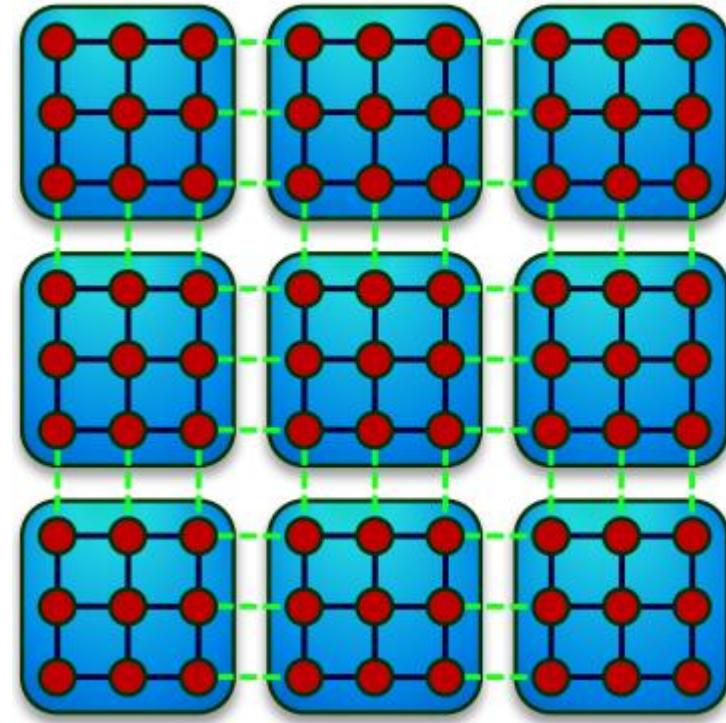
Titan Programming: Principle

- **Solving Problem Over Grid**

- E.g., finite-element system
- Simulate operation over time

- **Bulk Synchronous Model**

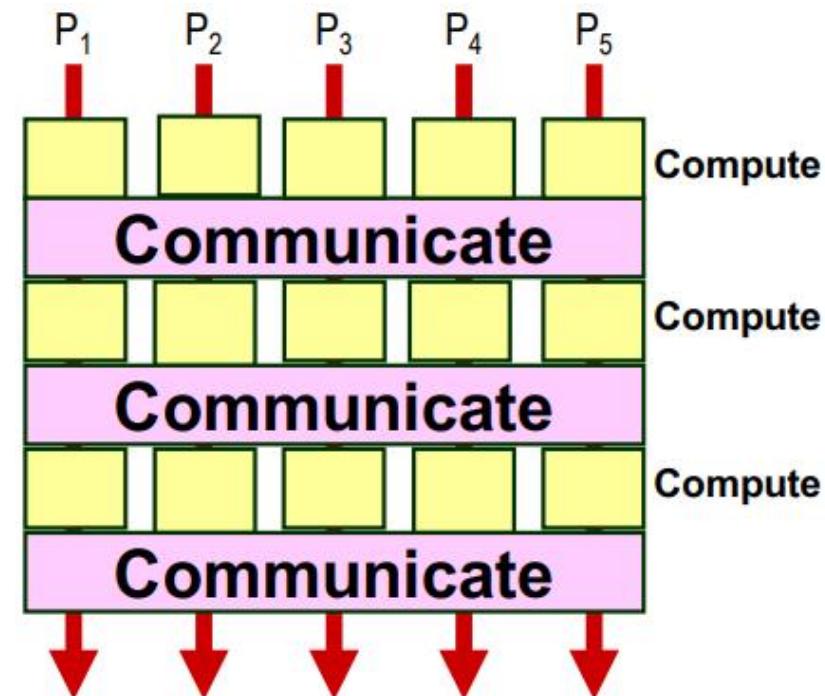
- Partition into Regions
 - p regions for p -node machine
- Map Region per Processor



Titan Programming: Principle (cont)

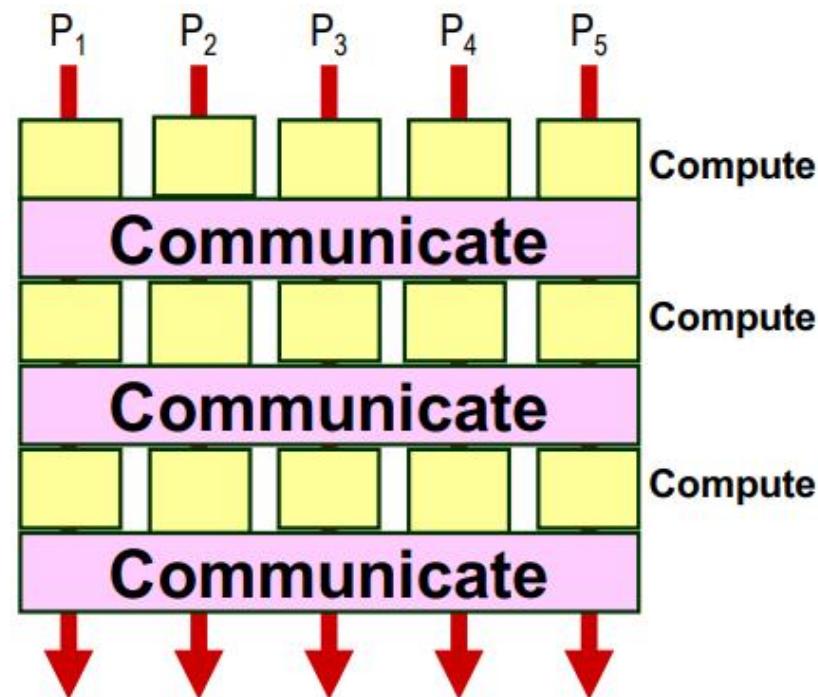
- Bulk Synchronous Model

- Map Region per Processor
- Alternate
 - All nodes compute behavior of region
 - Perform on GPUs
 - All nodes communicate values at boundaries



Bulk Synchronous Performance

- Limited by performance of slowest processor
- **Strive to keep perfectly balanced**
 - Engineer hardware to be highly reliable
 - Tune software to make as regular as possible
 - Eliminate “noise”
 - Operating system events
 - Extraneous network activity



General Architecture of the Sunway TaihuLight

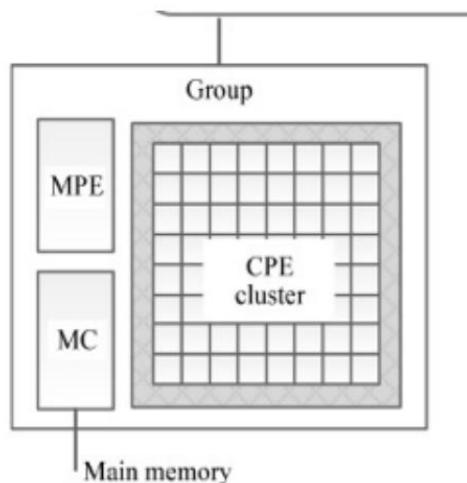
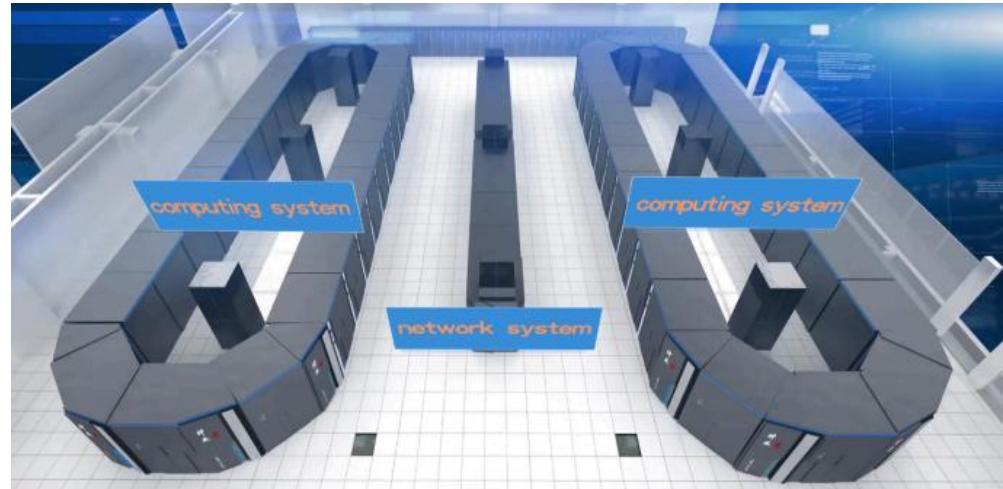
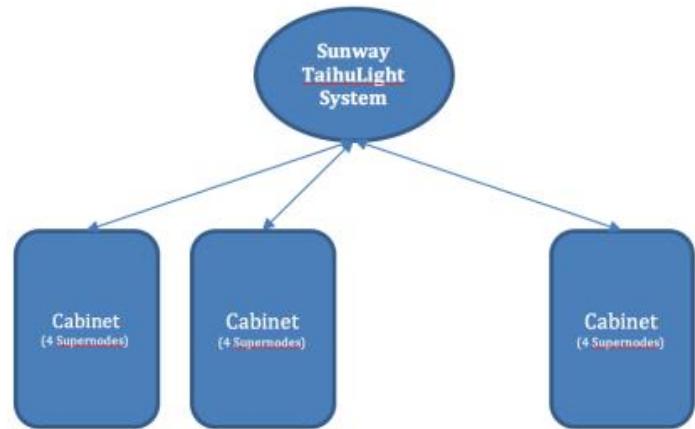


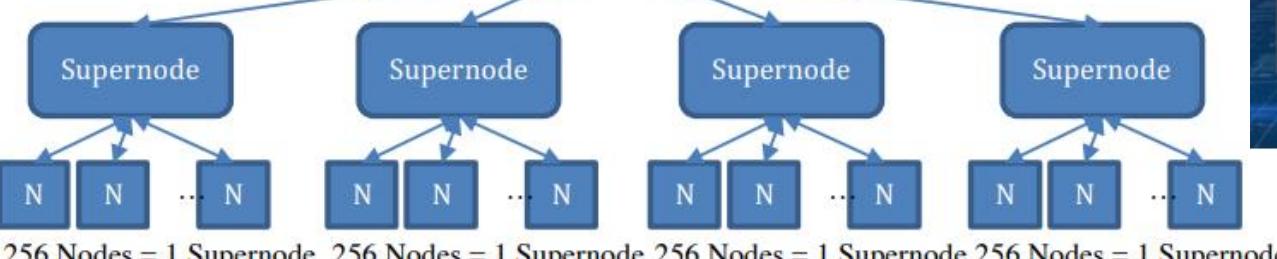
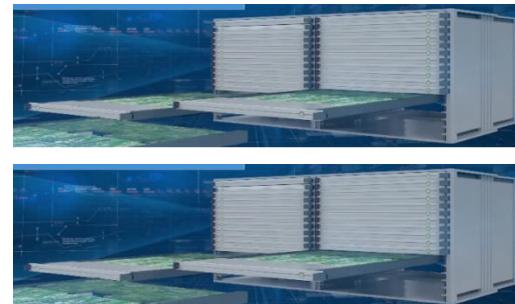
Figure 1: Core Group for Node



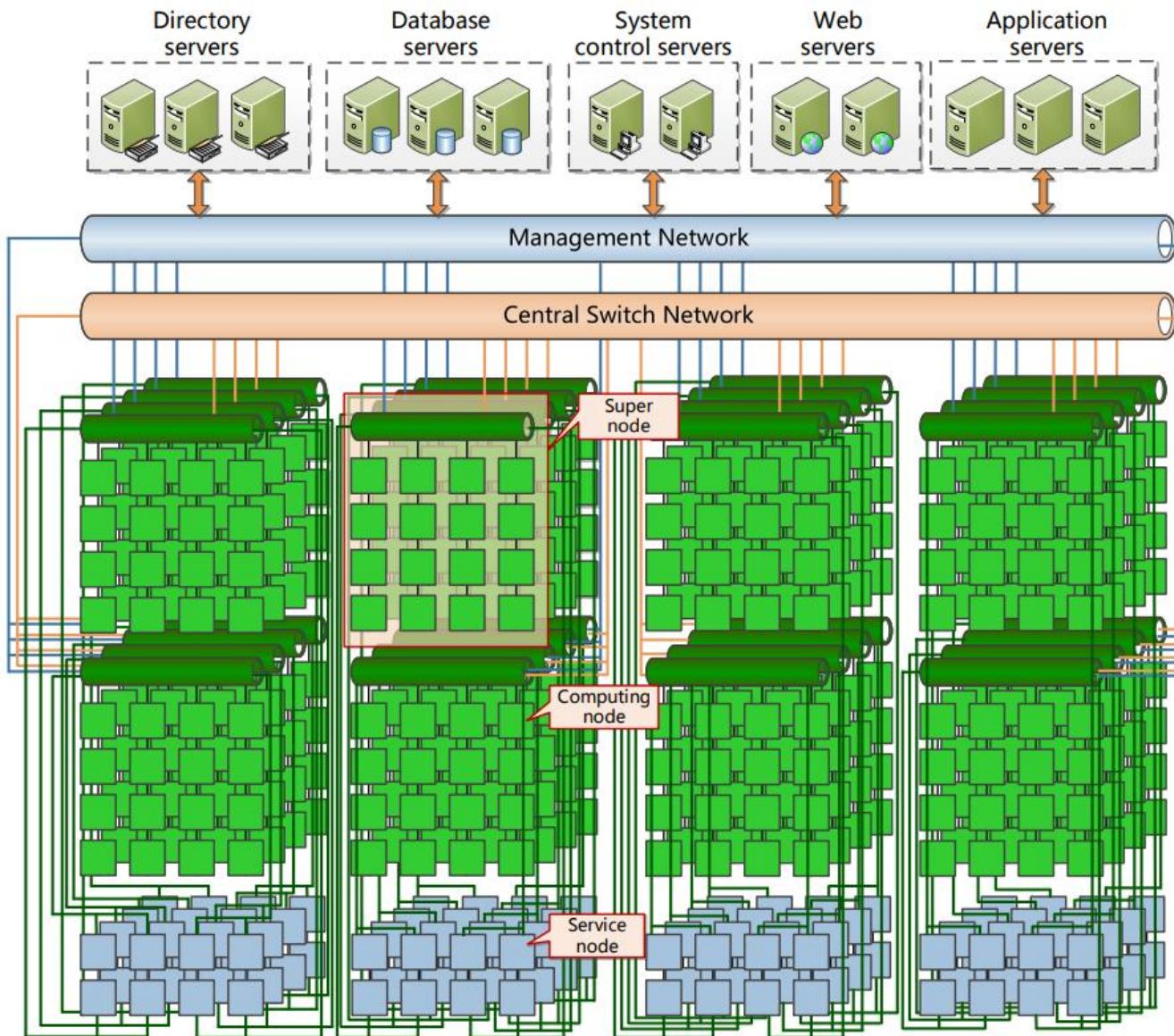
General Architecture of the Sunway TaihuLight



Cabinet
= 4
Super
nodes



General Architecture of the Sunway TaihuLight



Software Stack of the Sunway TaihuLight

Parallel Application

Parallel Program Development Environment

- IDE
- Parallel Debug
- Performance Monitor

Parallel Compiling Environment

- OpenACC

Compiling System

- C/C++, Fortran
- SIMD

Many-Core Basic Software

Basic Libs

- C Lib
- ACC Thread Lib
- Math Lib

Auto-vectorization

- C, C++, Fortran
- Loop Vectorization
- Code Optimization

Parallel OS Environment

- Job
 - Resource
 - Power
 - Network
- Fault Tolerant
 - System
 - Security

HPC Storage Management

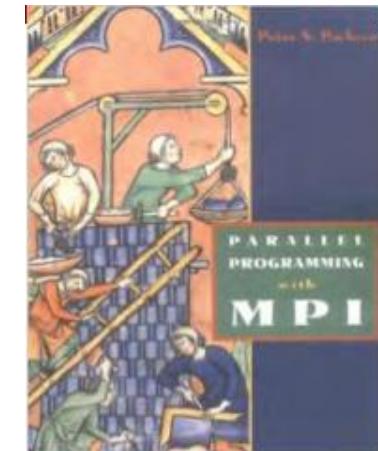
- SWGFS
- LWFS
- Storage Management Platform

Sunway TaihuLight System

Supercomputer Programming

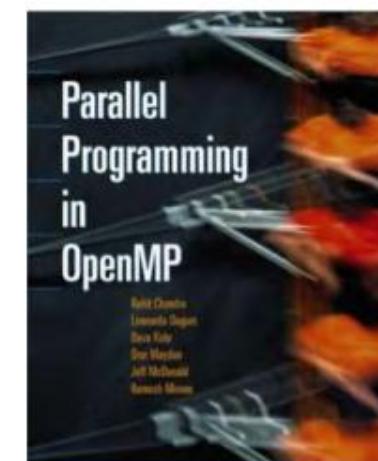
○ System Level

- Message-Passing Interface (MPI) supports node computation, synchronization and communication



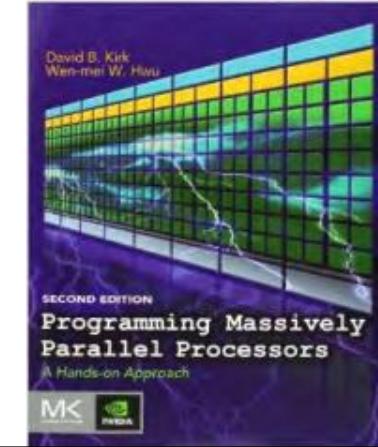
○ Node Level

- OpenMP supports thread-level operation of node CPU
- CUDA programming environment for GPUs
 - Performance degrades quickly if don't have perfect balance among memories and processors



○ Result

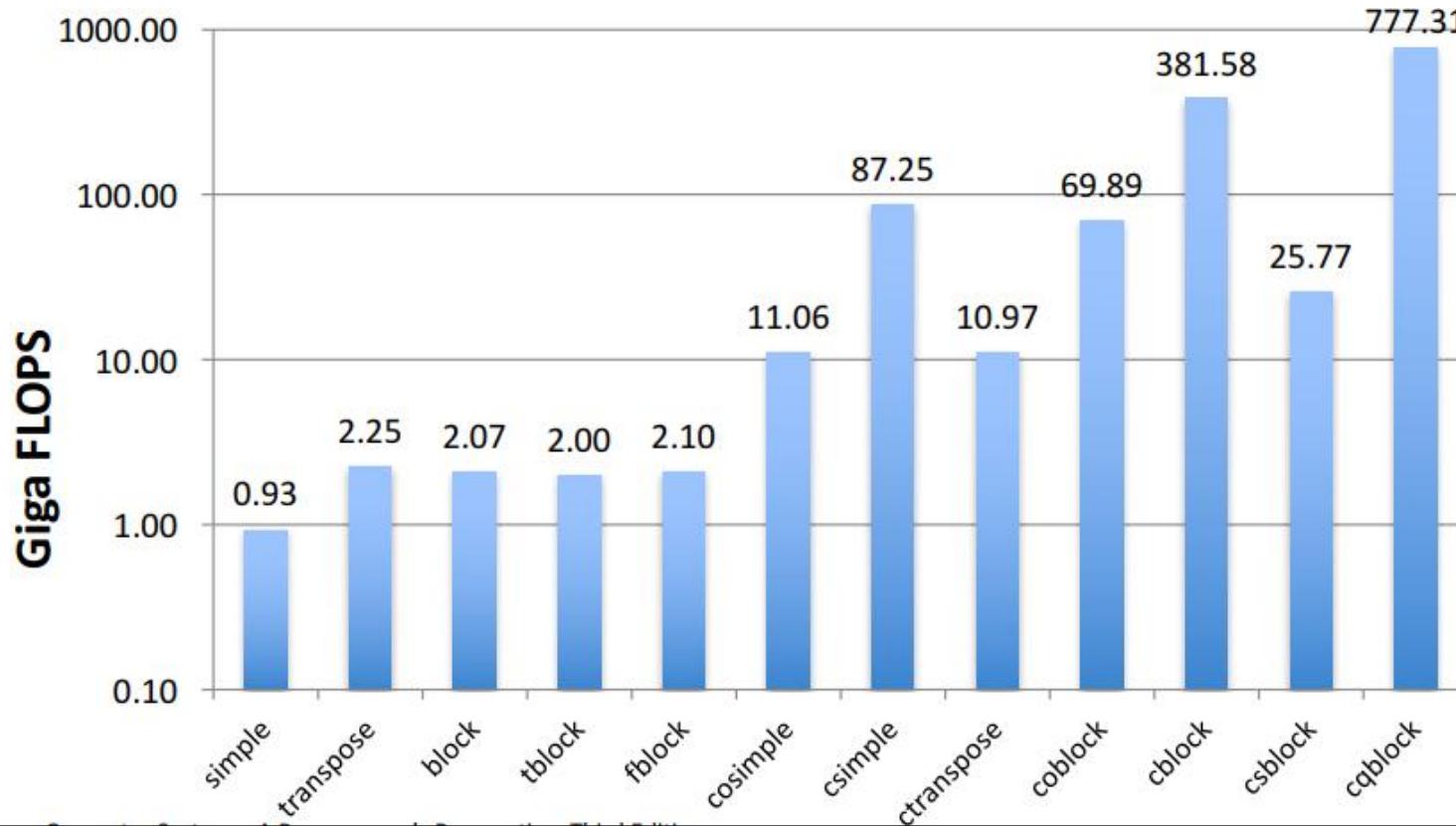
- Single program is complex combination of multiple programming paradigms
- Tend to optimize for specific hardware configuration



GPU Experience on Matrix Multiply

- Multiply two 1024×1024 matrices (MM)

- 2 $\times 10^9$ floating point operations
- Express performance in Giga FLOPS
- Program in CUDA and map onto nVidia GPU



Matrix Multiplication Progress

○ Versions

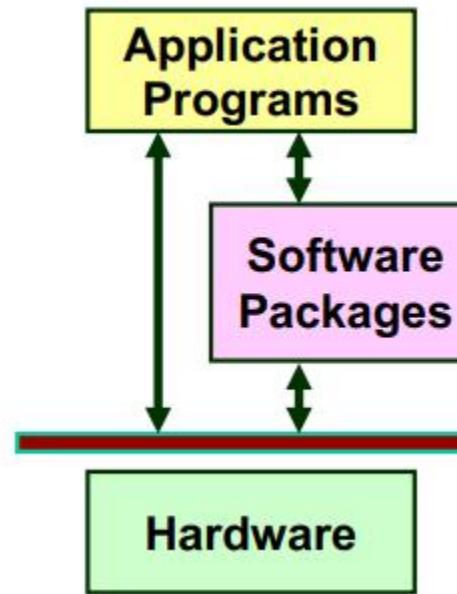
- Naive 1
- Simple parallel 11
- Blocking 70
- nVidia Example Code 388
- Reorient memory accesses 382
- Packed data access 777

○ Observations

- Progress is very nonlinear
 - Not even monotonic
- Requires increased understanding of how program maps onto hardware
- Becomes more specialized to specific hardware configuration

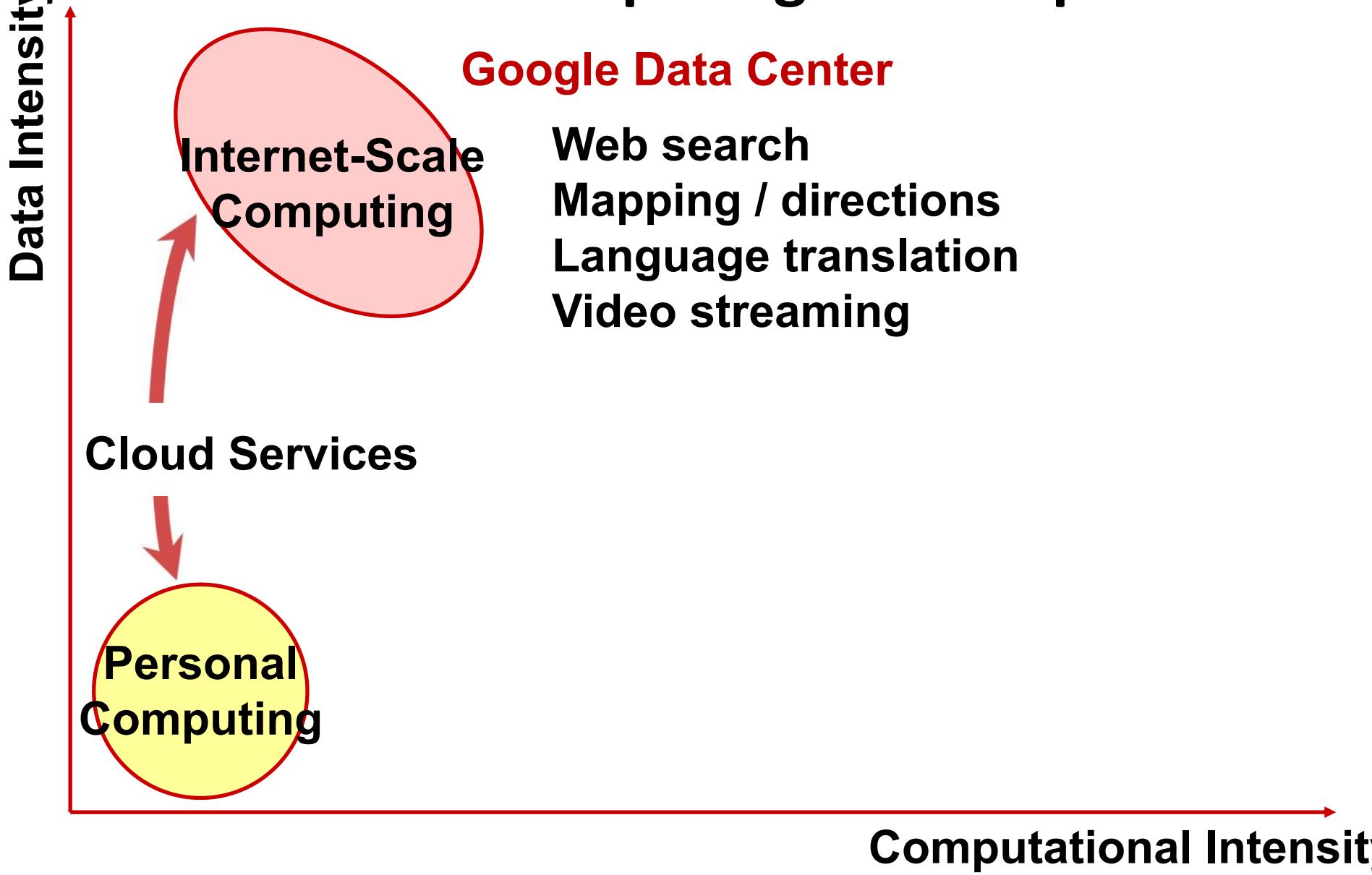
Supercomputer Programming Model

- Program on top of bare hardware
- Performance
 - Low-level programming to maximize node performance
 - Keep everything globally synchronized and balanced
- Reliability
 - Single failure causes major delay
 - Engineer hardware to minimize failures



**Machine-
Dependent
Programmi
ng Mode**

Data-Intensive Computing Landscape



Internet Computing

o Web Search

- Aggregate text data from across WWW
- No definition of correct operation
- Do not need real-time updating

o Mapping Services

- Huge amount of (relatively) static data
- Each customer requires individualized computation



o Online Documents

- Must be stored reliably
- Must support real-time updating
- (Relatively) small data volumes

Other Data-Intensive Computing Applications

- **Wal-Mart**

- 267 million items/day, sold at 6,000 stores
- HP built them 4 PB data warehouse
- Mine data to manage supply chain, understand market trends, formulate pricing strategies



- **LSST**

- Chilean telescope will scan entire sky every 3 days
- A 3.2 gigapixel digital camera
- Generate 30 TB/day of image data



Data-Intensive Application Characteristics

- **Diverse Classes of Data**

- Structured & unstructured
- High & low integrity requirements

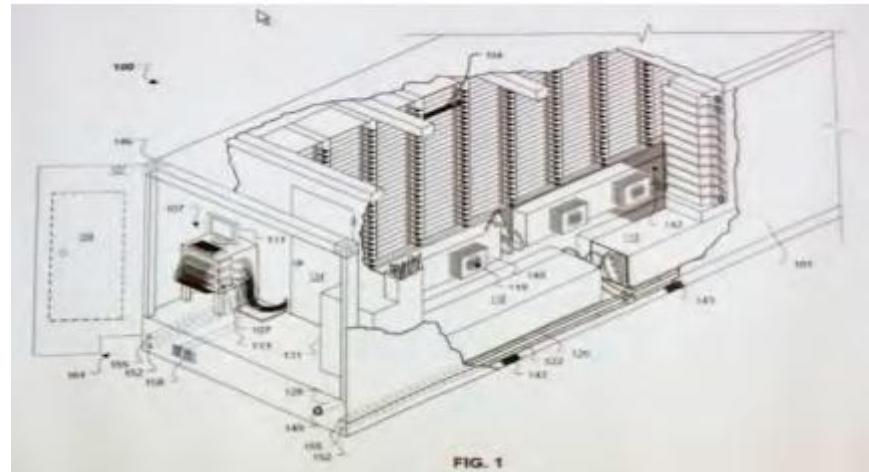
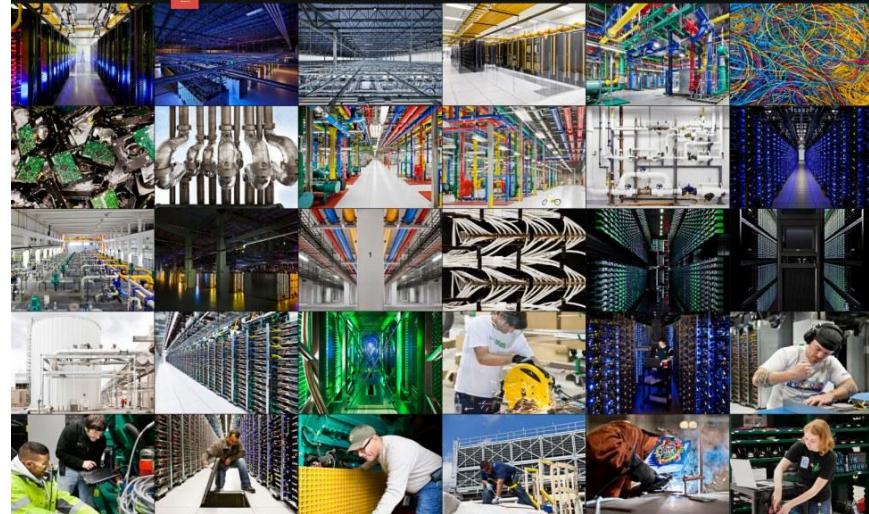
- **Diverse Computing Needs**

- Localized & global processing
- Numerical & non-numerical
- Real-time & batch processing

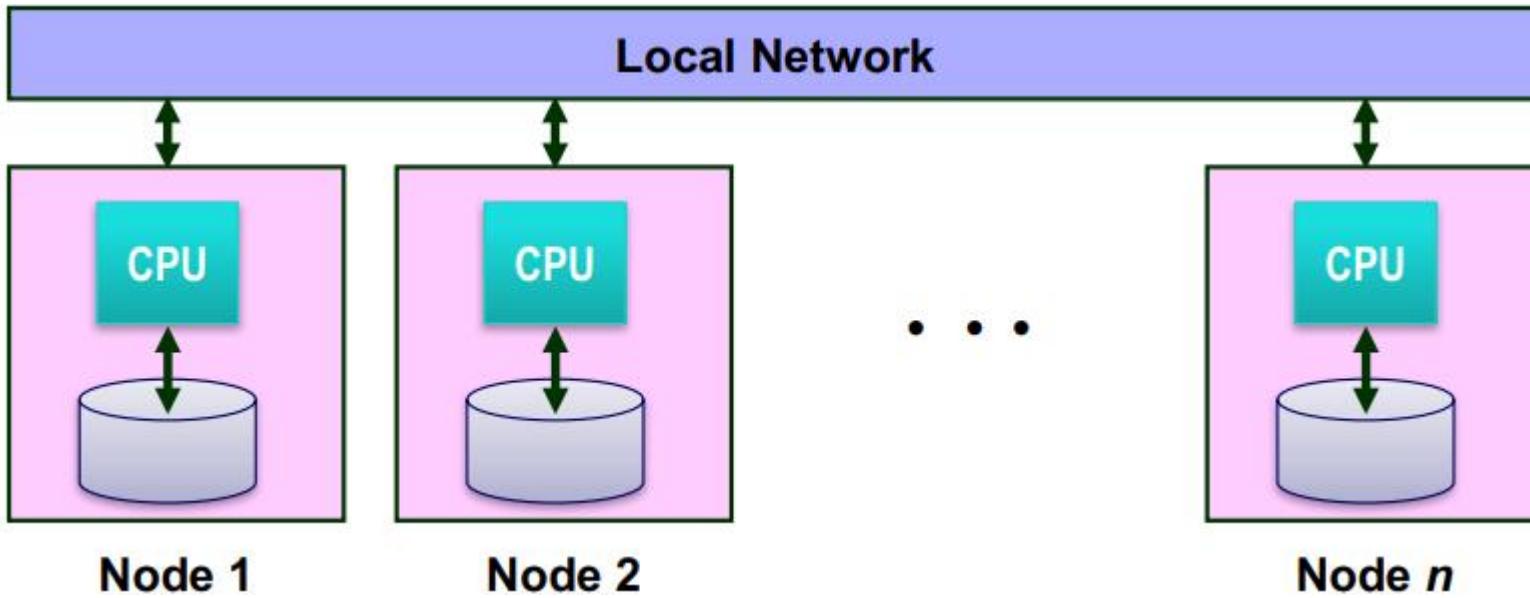
Google Data Centers

○ Dalles, Oregon

- Hydroelectric power @ 2¢ / KW Hr
- 50 Megawatts
- Enough to power 60,000 homes
- Engineered for low cost, modularity & power efficiency
- Container: 1160 server nodes, 250KW



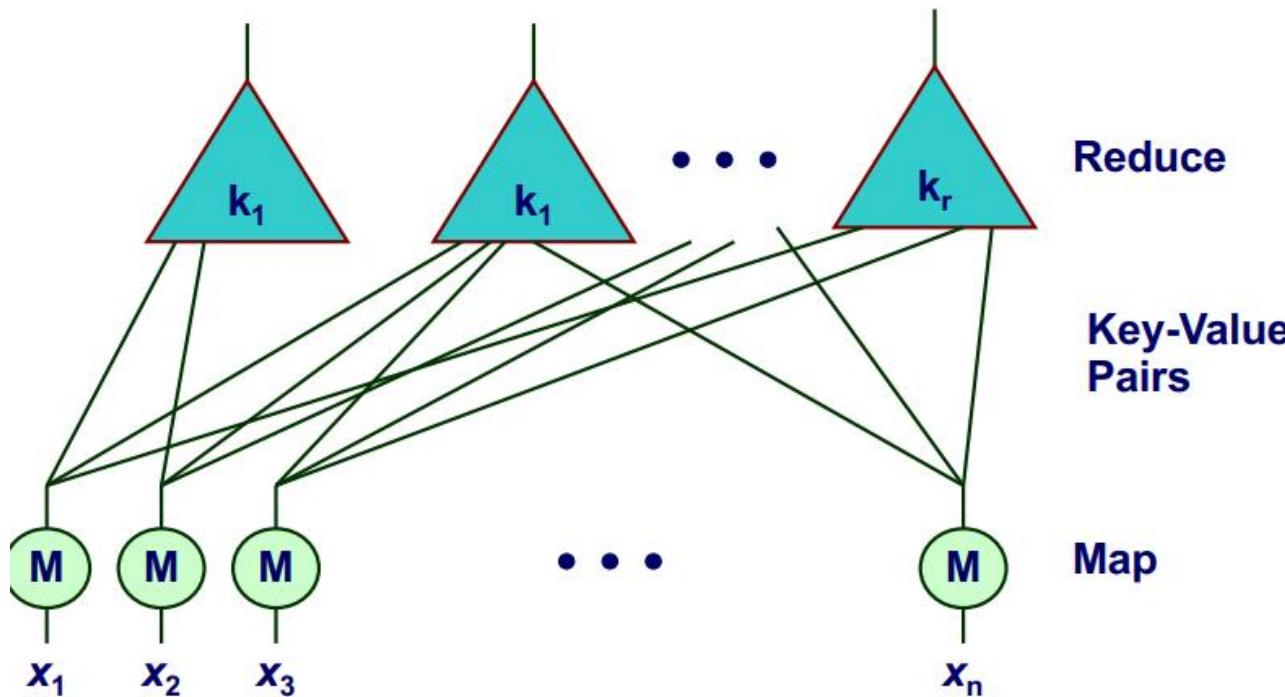
Google Cluster



- Typically 1,000–2,000 nodes
- Node Contains
 - 2 multicore CPUs
 - 2 disk drives
 - DRAM



Map/Reduce Programming Model



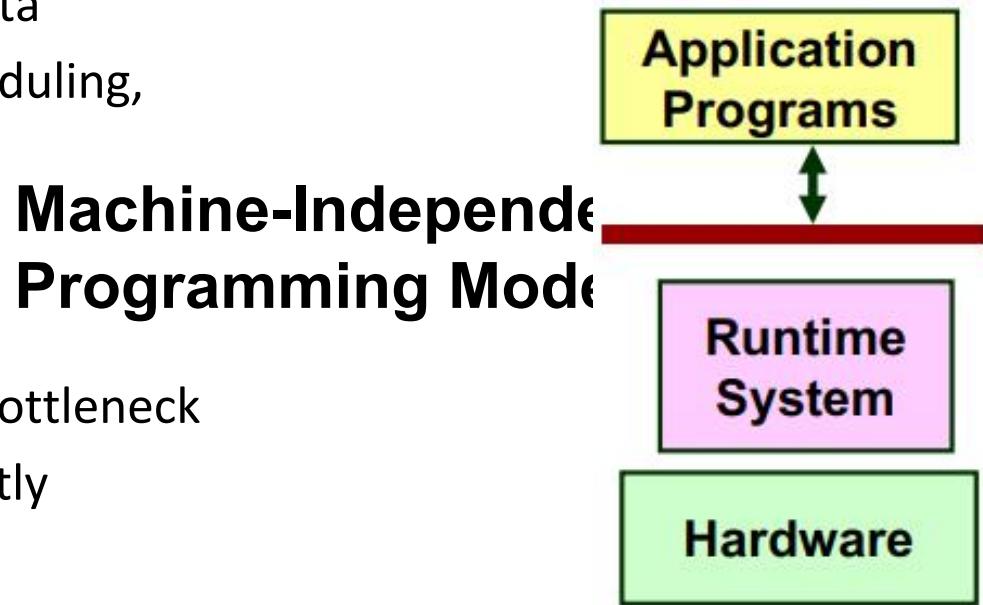
- **Map computation across many objects**
 - E.g., 10^{10} Internet web pages
- **Aggregate results in many different ways**
- **System deals with issues of resource allocation & reliability**

Cluster Programming Model

- Application programs written in terms of high-level operations on data
- Runtime system controls scheduling, load balancing, ...

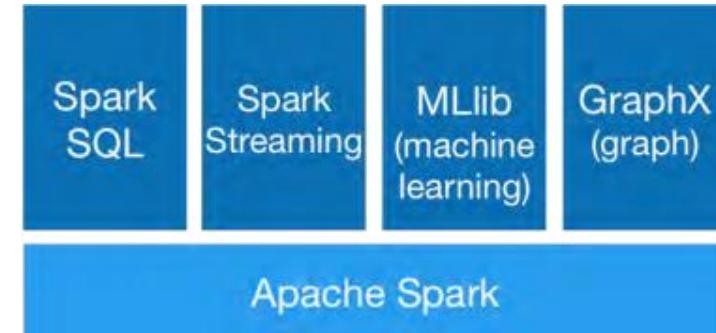
○ Scaling Challenges

- Centralized scheduler forms bottleneck
- Copying to/from disk very costly
- Hard to limit data movement
 - Significant performance factor



Some Typical Programming Systems

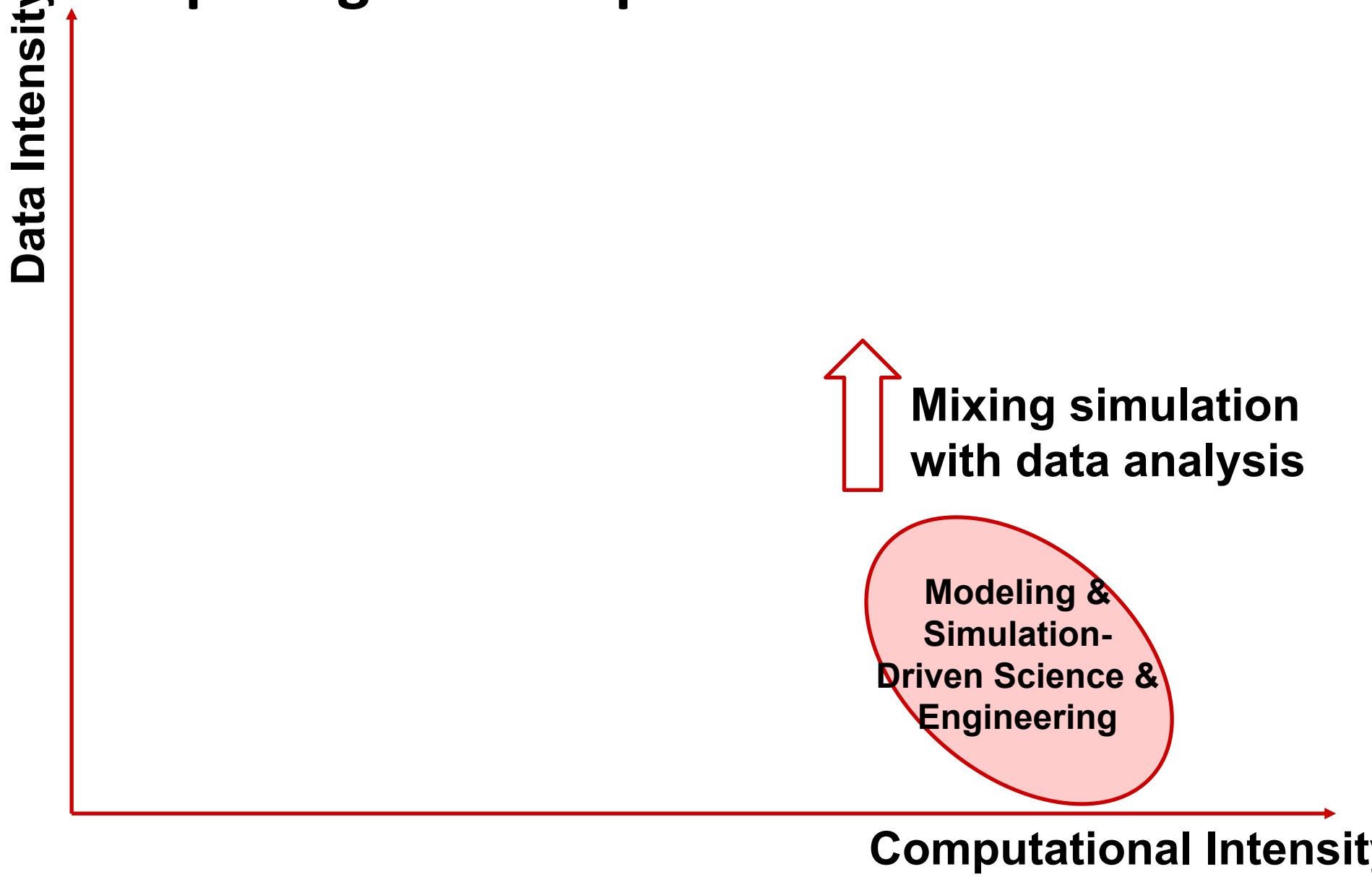
- **Spark Project**
- **at U.C., Berkeley**
- **Grown to have large open source community**



- **GraphLab**
- **Started as project at CMU by Carlos Guestrin**
- **Environment for describing machine-learning algorithms**
 - Sparse matrix structure described by graph
 - Computation based on updating of node values



Computing Landscape Trends



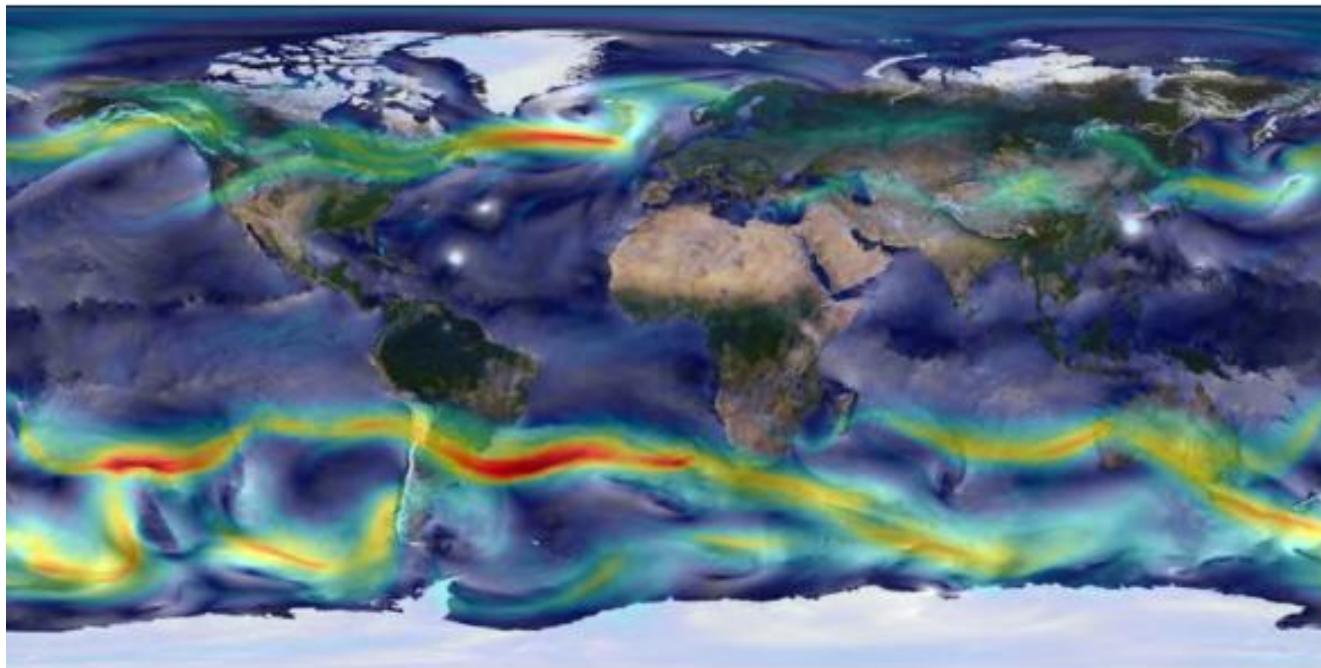
Combining Simulation with Real Data

- **Limitations**

- Simulation alone: Hard to know if model is correct
- Data alone: Hard to understand causality & “what if”

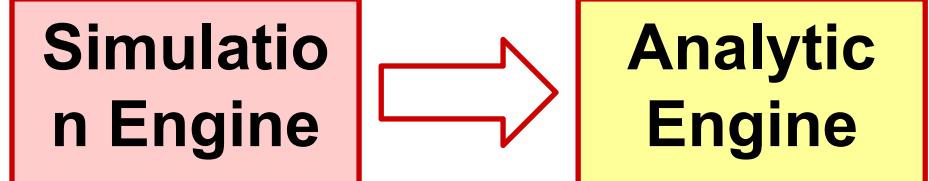
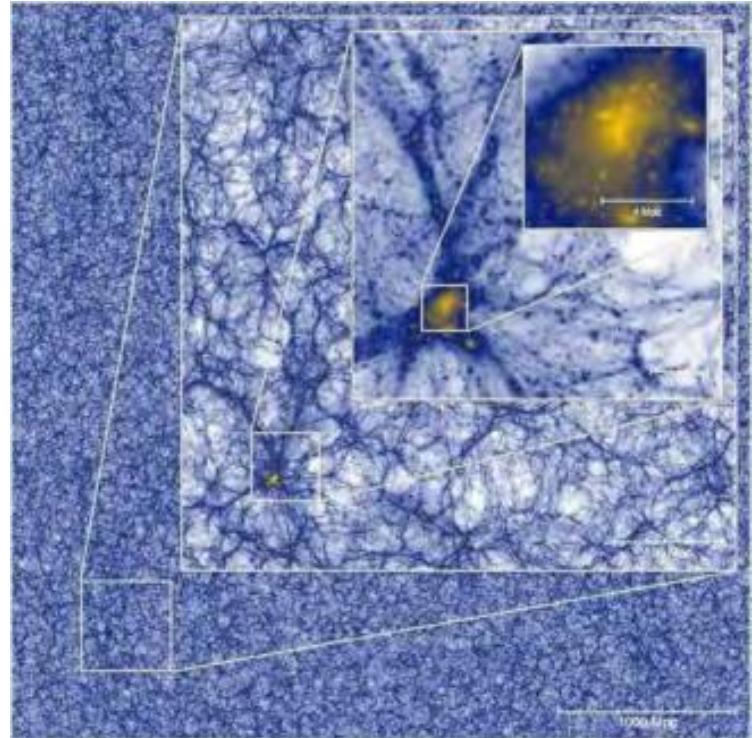
- **Combination**

- Check and adjust model during simulation

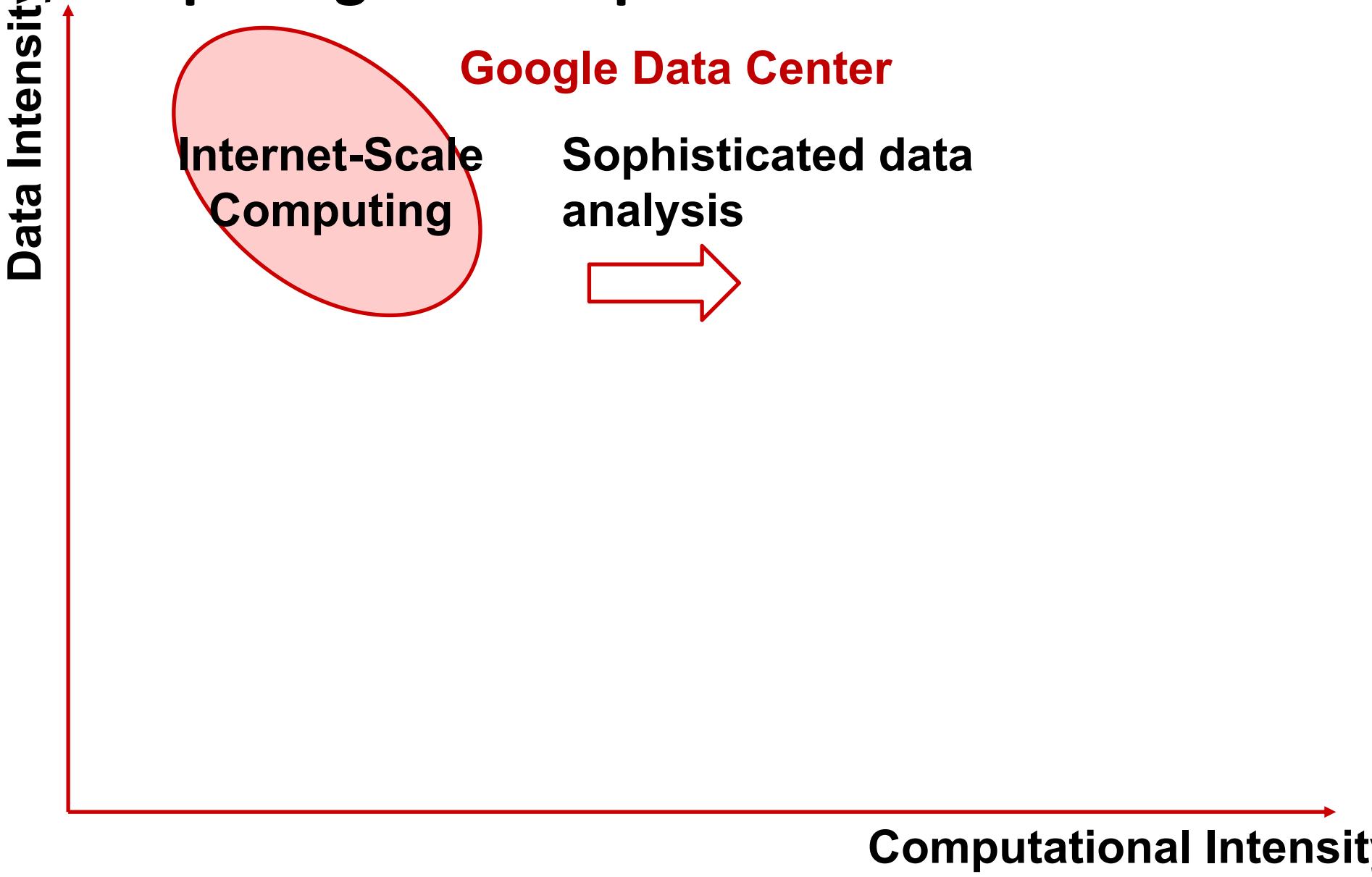


Real-Time Analytics

- **Millenium XXL Simulation (2010)**
 - 3×10^9 particles
 - Simulation run of 9.3 days on 12,228 cores
 - 700TB total data generated
 - Save at only 4 time points
 - 70 TB
 - Large-scale simulations generate large data sets
- **What If?**
 - Could perform data analysis while simulation is running

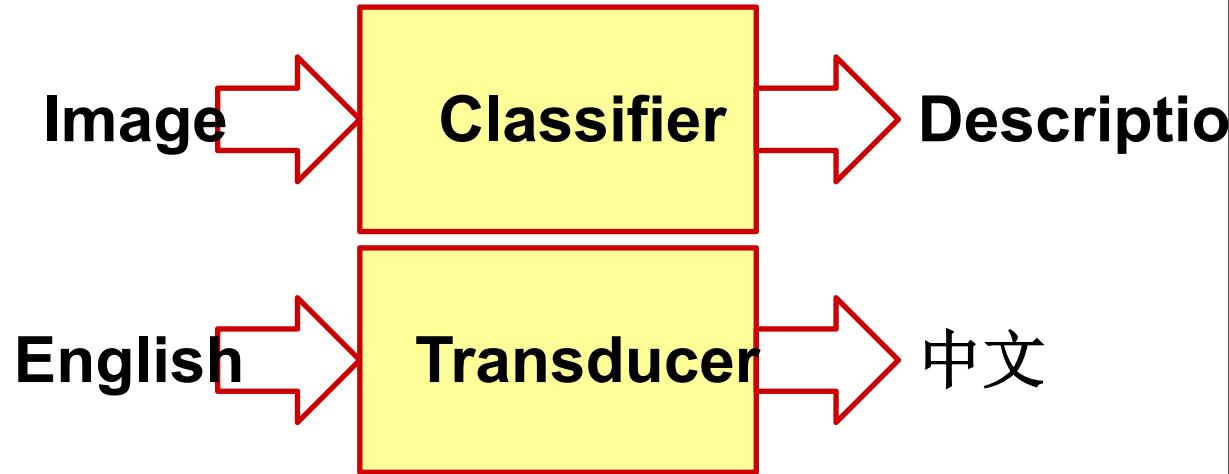


Computing Landscape Trends



Example Analytic Applications

Microsoft Project Adam



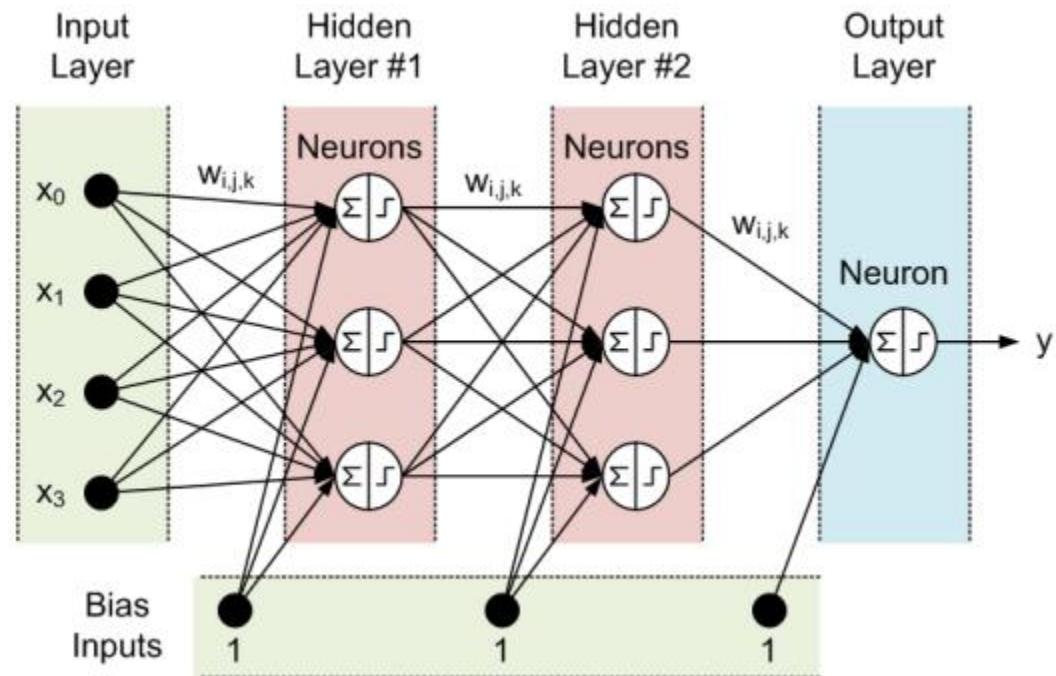
Google Translate

A screenshot of the Google Translate web interface. At the top, there are tabs for 'Text', 'Images', 'Documents', and 'Websites'. Below that, a 'Detect language' dropdown is set to 'English', with options for Spanish and French. The main area has a large input text box on the left and a 'Translation' output box on the right. A progress bar at the bottom of the input box shows '0 / 5,000'. In the bottom right corner, there is a link 'Send feedback'.

Data Analysis with Deep Neural Networks

- Task:

- Compute classification of set of input signals



- Training

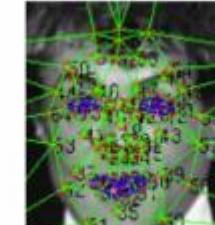
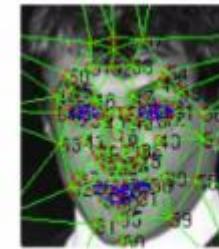
- Use many training samples of form input / desired output
- Compute weights that minimize classification error

- Operation

- Propagate signals from input to output

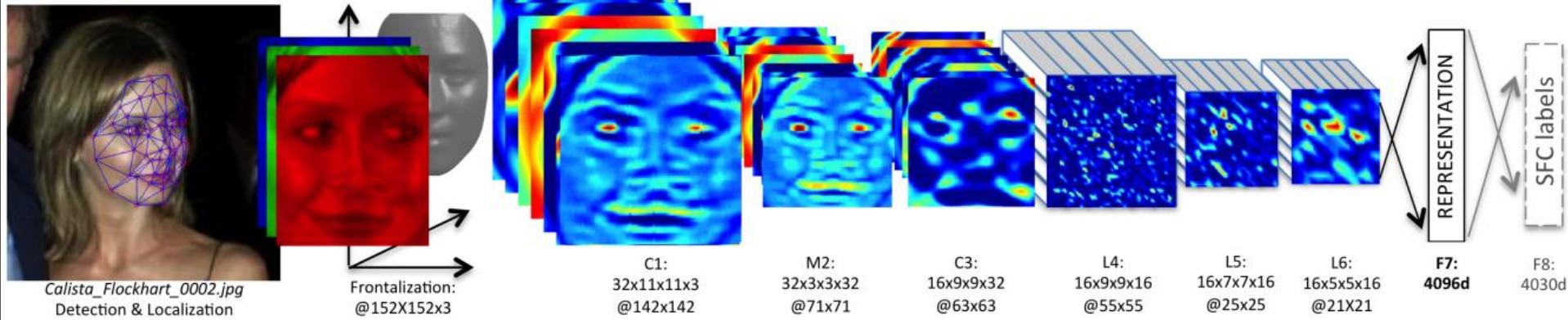
DNN Application Example

- Facebook DeepFace Architecture



2D-alignment

3D-alignment



Training DNNs



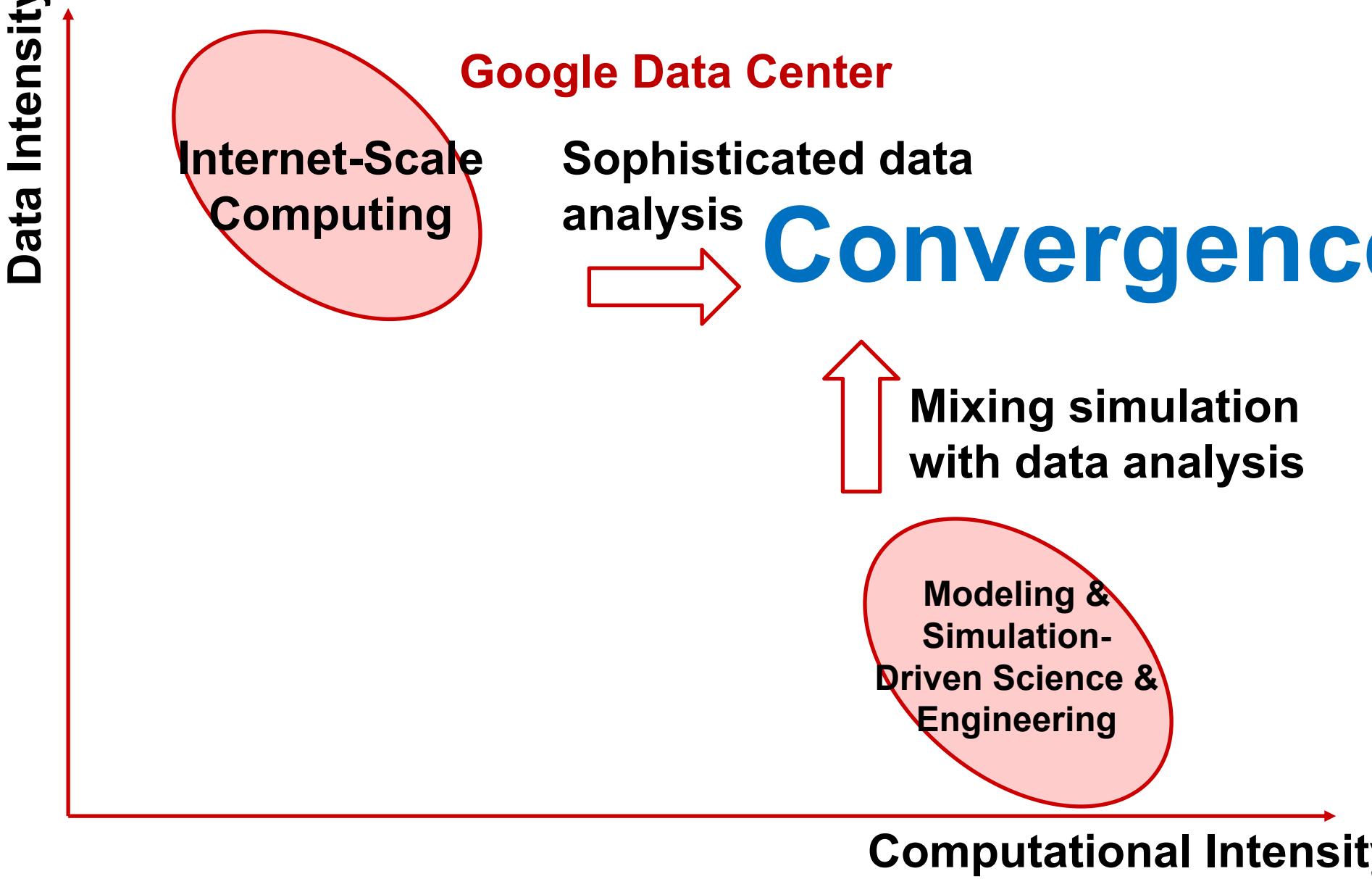
○ Characteristics

- Iterative numerical algorithm
- Regular data organization

○ Project Adam Training

- 2B connections
- 15M images
- 62 machines
- 10 days

Trends



Challenges for Convergence

Supercomputers

- Customized
- Optimized for reliability

Run-Time System

- Source of “noise”
- Static scheduling

Application Programming

- Low-level, processor-centric model

Data Center Clusters

Hardware

- Consumer grade
- Optimized for low cost

- Provides reliability
- Dynamic allocation

- High level, data-centric model

Computation/Data Convergence

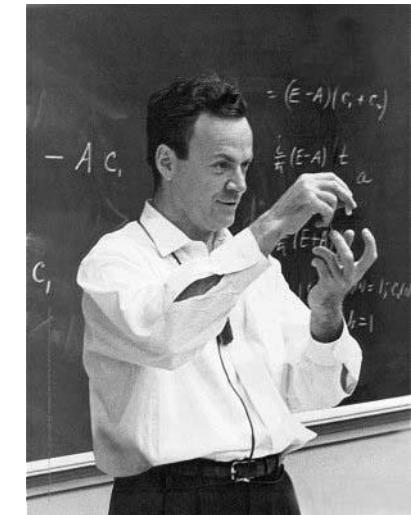
- **Two Important Classes of Large-Scale Computing**
 - Computationally intensive supercomputing
 - Data intensive processing
 - Internet companies + many other applications
- **Followed Different Evolutionary Paths**
 - Supercomputers: Get maximum performance from available hardware
 - Data center clusters: Maximize cost/performance over variety of datacentric tasks
 - Yielded different approaches to hardware, runtime systems, and application programming
- **A Convergence Would Have Important Benefits**
 - Computational and data-intensive applications
 - But, not clear how to do it

Future of Computing

QUANTUM COMPUTING

Introduction

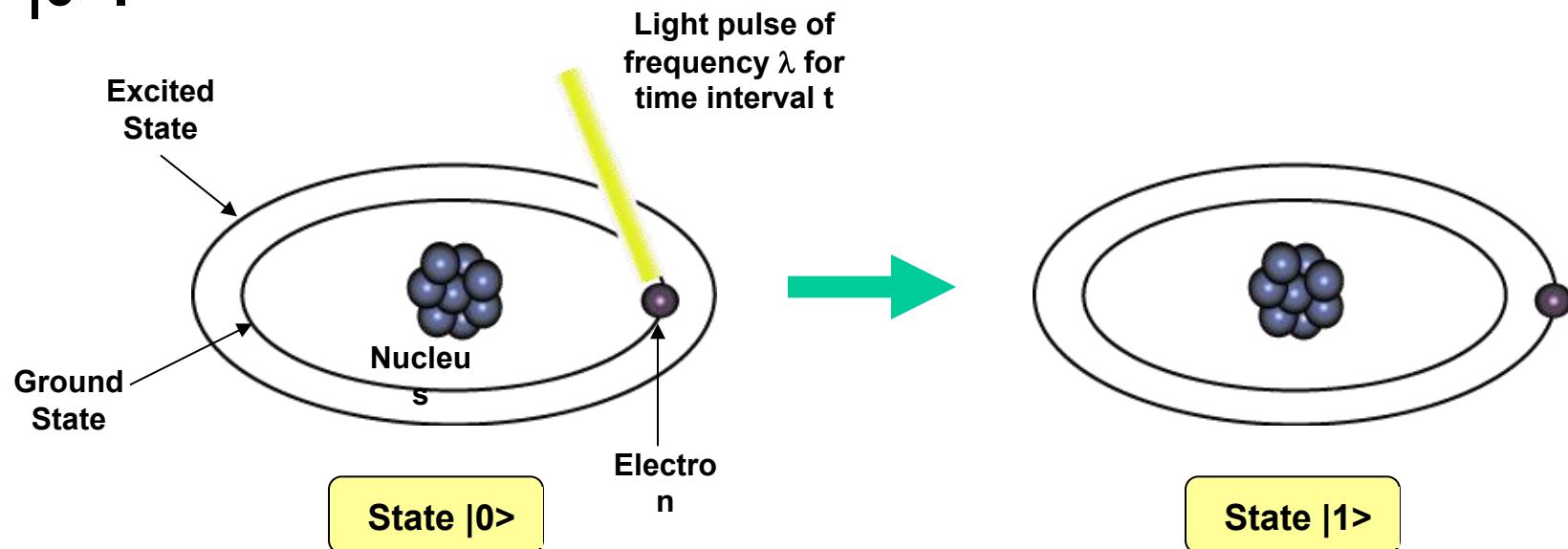
- “I think I can safely say that nobody understands quantum mechanics” - Feynman
- 1982 - Feynman proposed the idea of creating machines based on the laws of quantum mechanics instead of the laws of classical physics.
- 1985 - David Deutsch developed the quantum turing machine, showing that quantum circuits are universal.
- 1994 - Peter Shor came up with a quantum algorithm to factor very large numbers in polynomial time.
- 1997 - Lov Grover develops a quantum search algorithm with $O(\sqrt{N})$ complexity



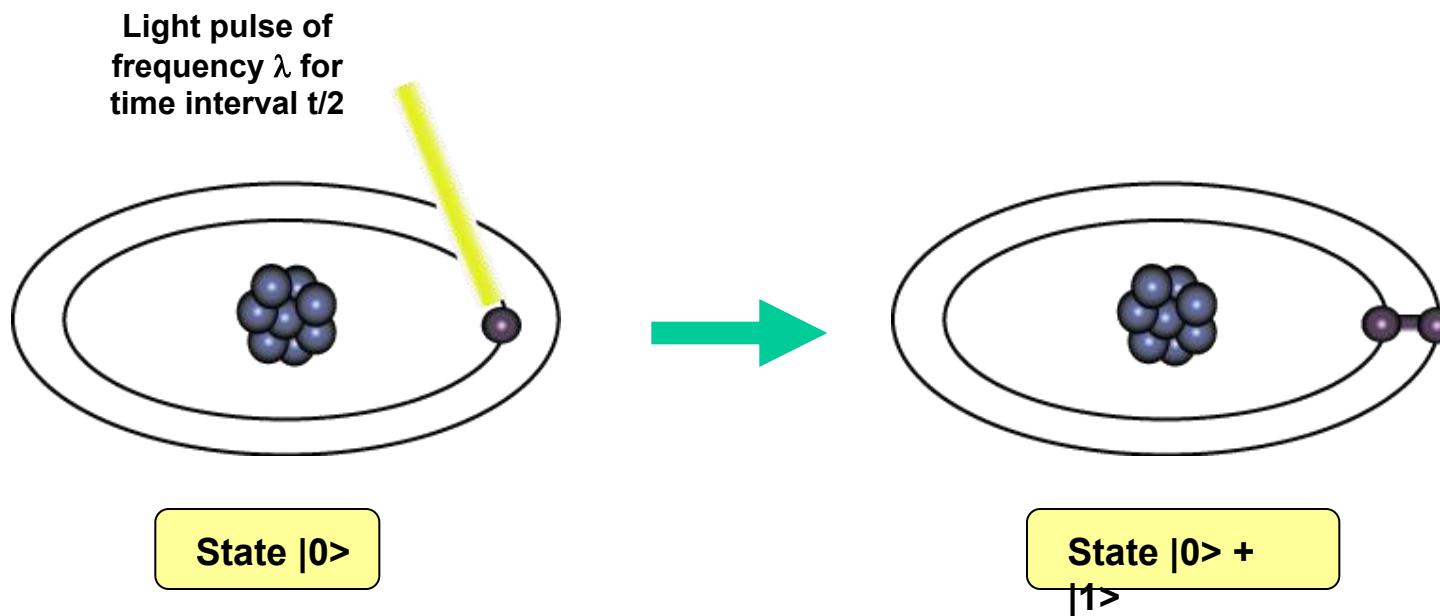
Representation of Data - Qubits

A bit of data is represented by a single atom that is in one of two states denoted by $|0\rangle$ and $|1\rangle$. A single bit of this form is known as a *qubit*.

A physical implementation of a qubit could use the two energy levels of an atom. An excited state representing $|1\rangle$ and a ground state representing $|0\rangle$.



Representation of Data - Superposition

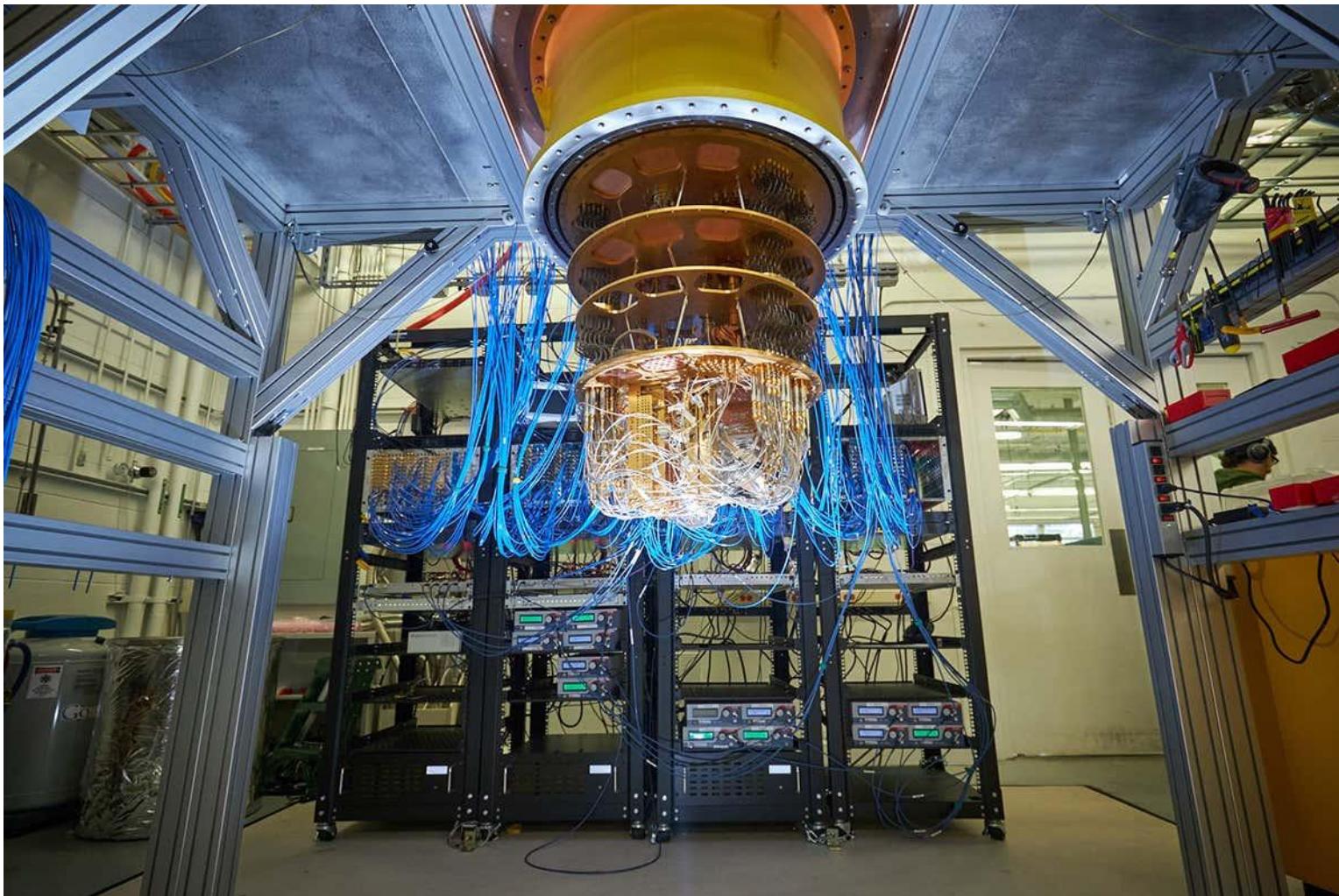


Consider a 3 bit qubit register. An equally weighted superposition of all possible states would be denoted by:

$$|\psi\rangle = \frac{1}{\sqrt{8}}|000\rangle + \frac{1}{\sqrt{8}}|001\rangle + \dots + \frac{1}{\sqrt{8}}|111\rangle$$

Quantum Computing Machine

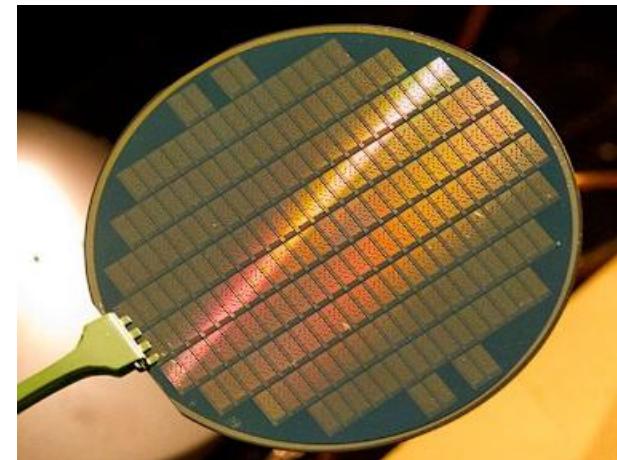
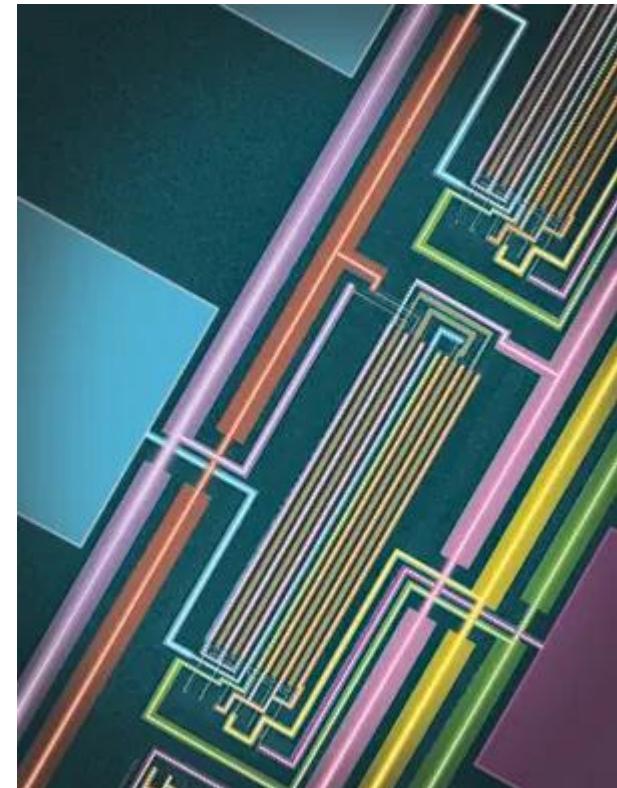
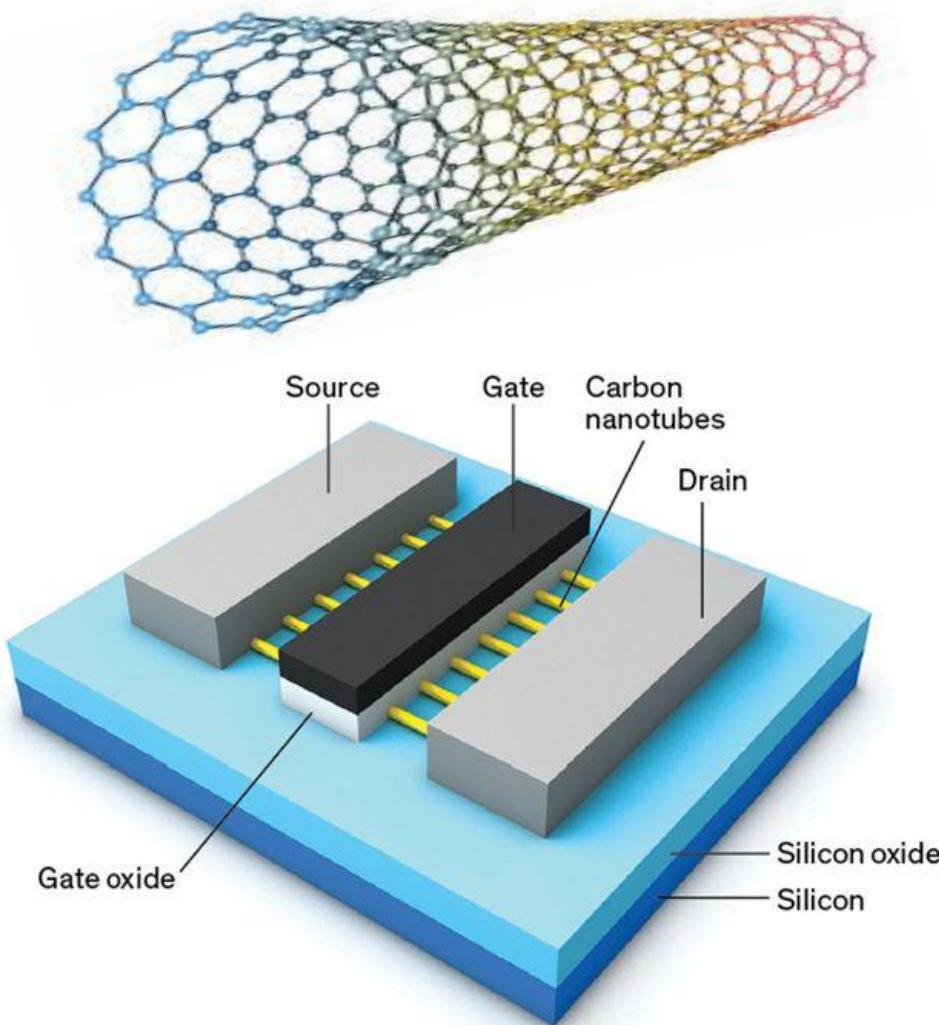
- Google's Sycamore quantum computer



Future of Computing

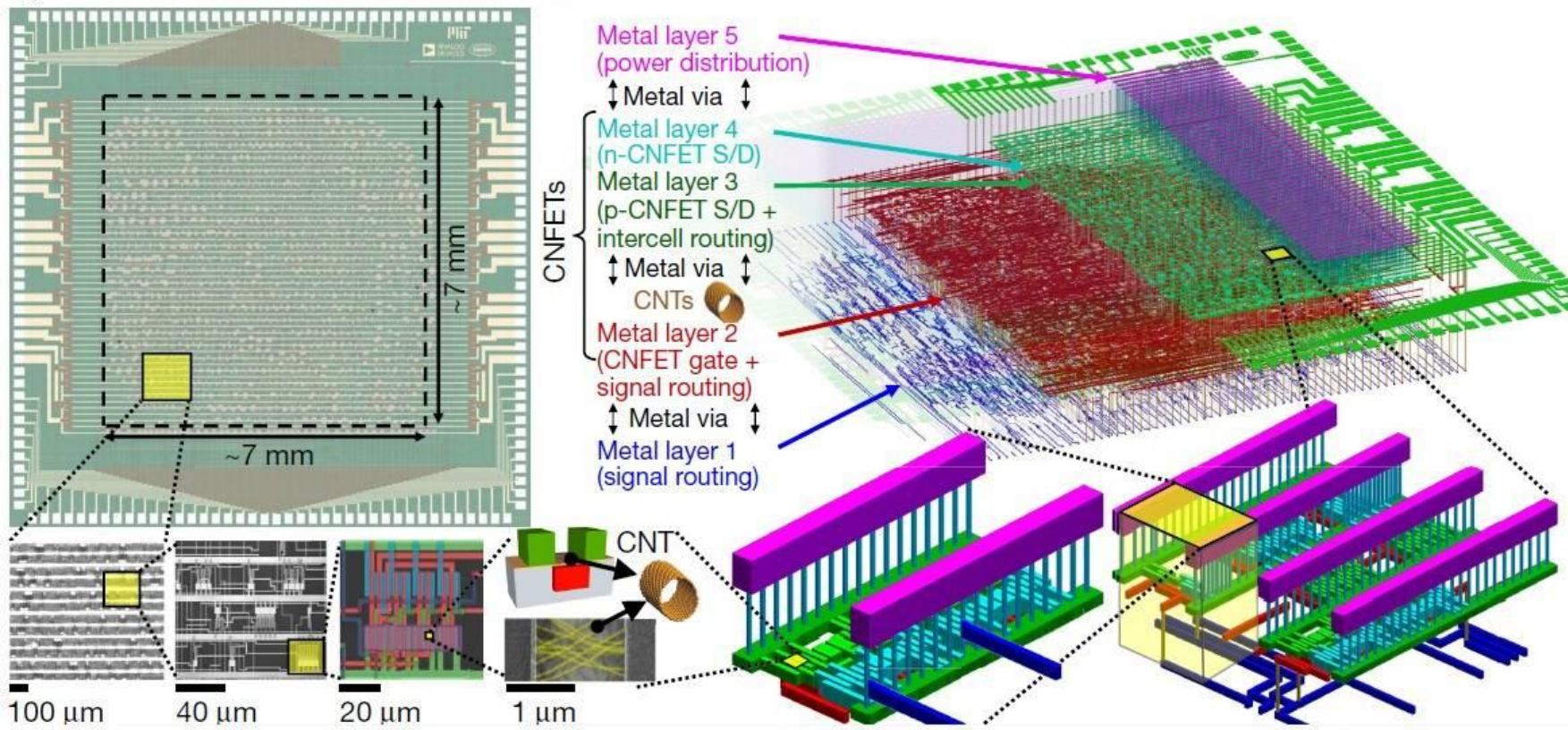
CARBON NANOTUBE COMPUTER

CNT Transistor & Circuit



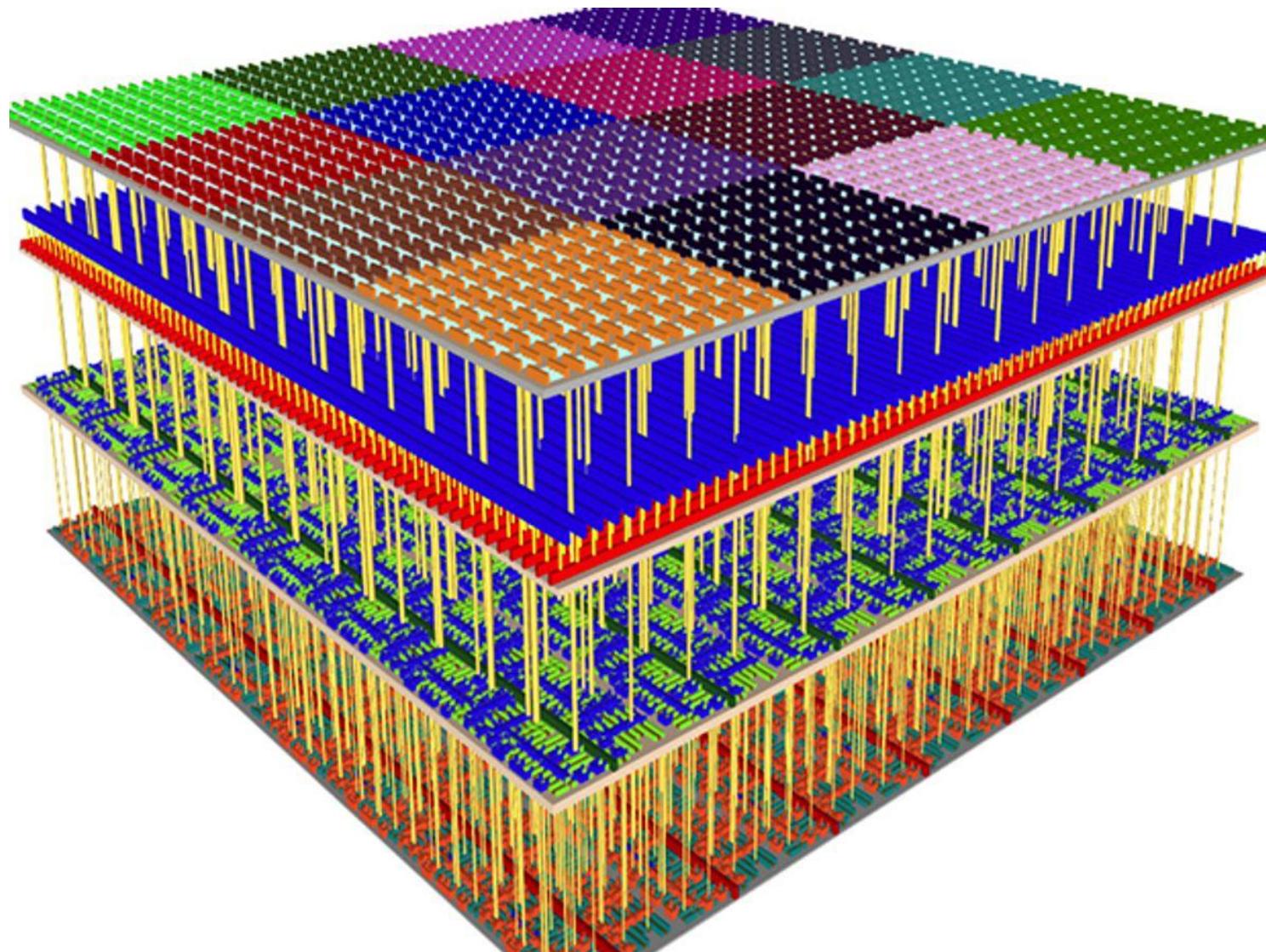
CNT Microprocessor

- CNT are only about 1–2 nm in diameter
- Excellent electronic properties



RV16X-NANO. Credit: Nature 572, 595–602 (2019)

3D CNT Computer



Three-dimensional integration of nanotechnologies for
high performance computing

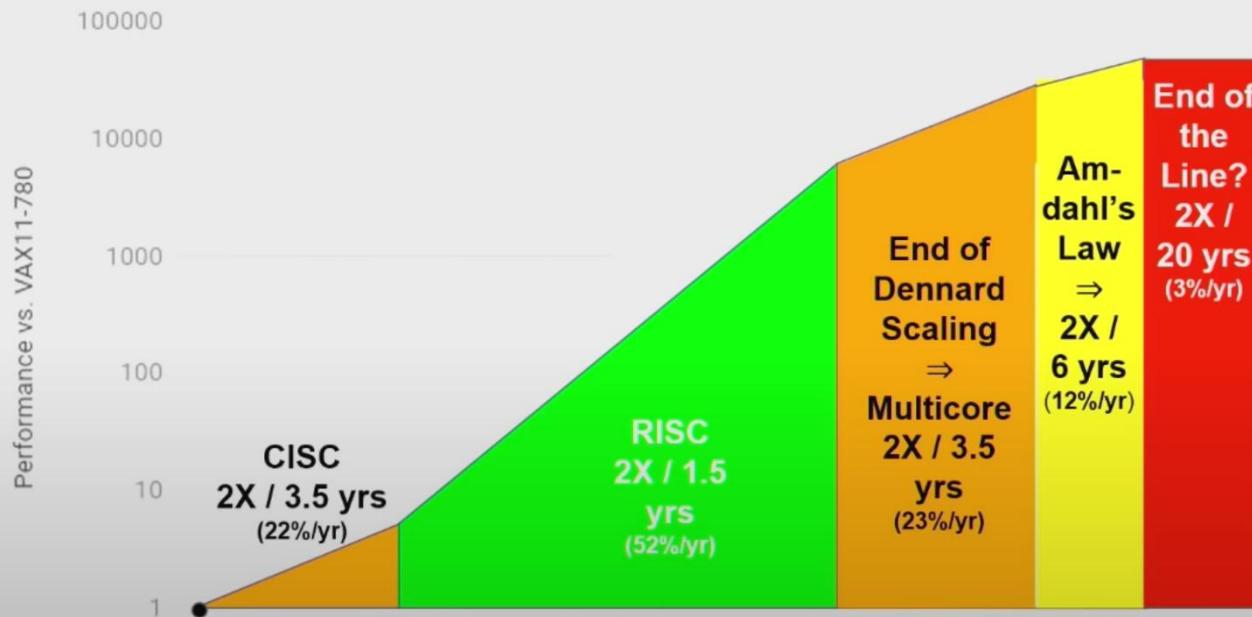
Future of Computing

DOMAIN SPECIFIC ARCHITECTURE

A New Golden Age for Computer Architecture

End of Growth of Single Program Speed?

40 years of Processor Performance



Based on SPECintCPU. Source: John Hennessy and David Patterson, Computer Architecture: A Quantitative Approach, 6/e. 2018

A New Golden Age for Computer Architecture

What Opportunities Left?

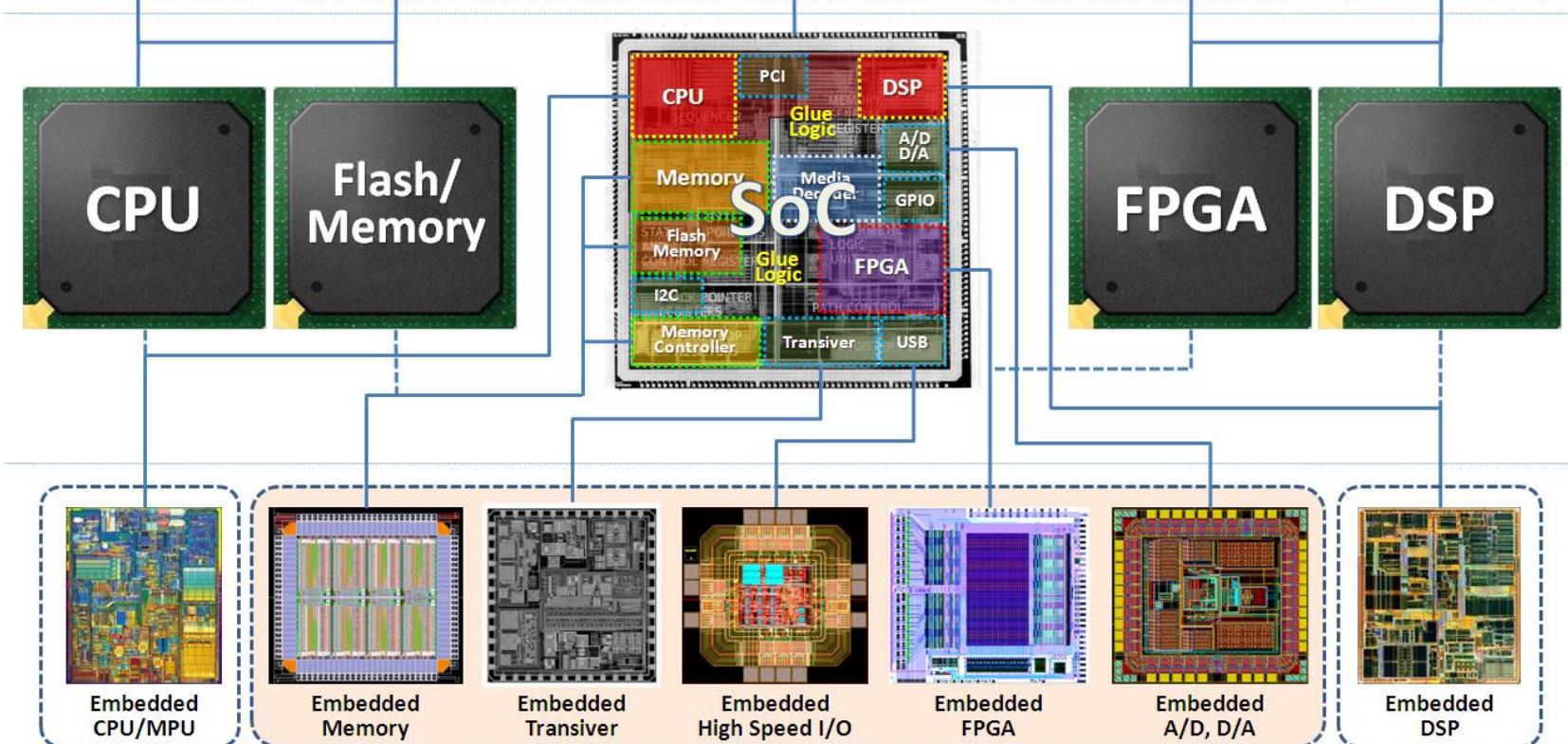
- SW-centric
 - Modern scripting languages are interpreted, dynamically-typed and encourage reuse
 - Efficient for programmers but not for execution
- HW-centric
 - Only path left is *Domain Specific Architectures*
 - Just do a few tasks, but extremely well
- Combination
 - Domain Specific Languages & Architectures

A New Golden Age for Computer Architecture

Part III: DSL/DSA Summary

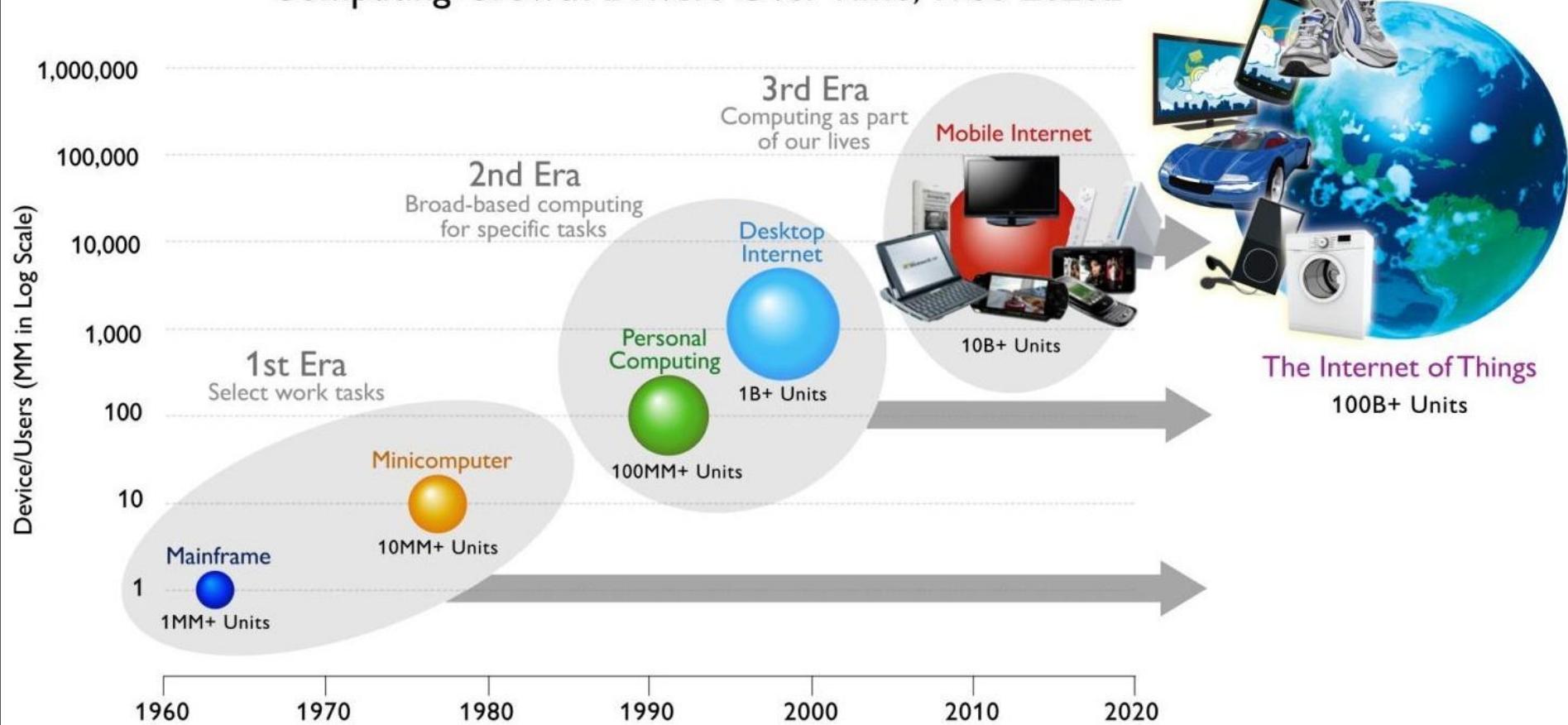
- Lots of opportunities
- But, new approach to computer architecture is needed.
- The Renaissance computer architecture team is vertically integrated. Understands:
 - Applications
 - DSLs and related compiler technology
 - Principles of architecture
 - Implementation technology
- Everything old is new again!

The range of microprocessors is expanding

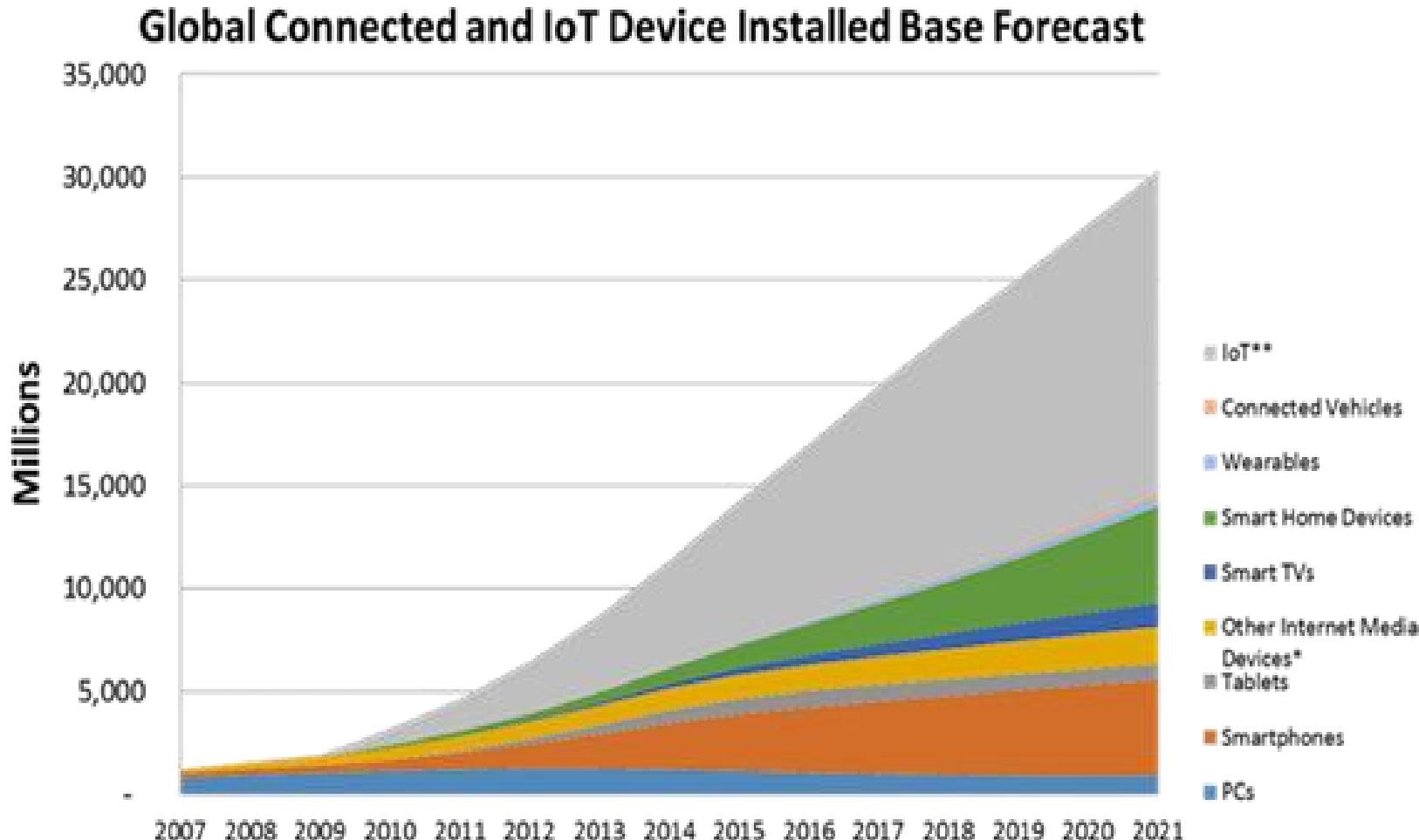


Post-PC Era

Computing Growth Drivers Over Time, 1960-2020E



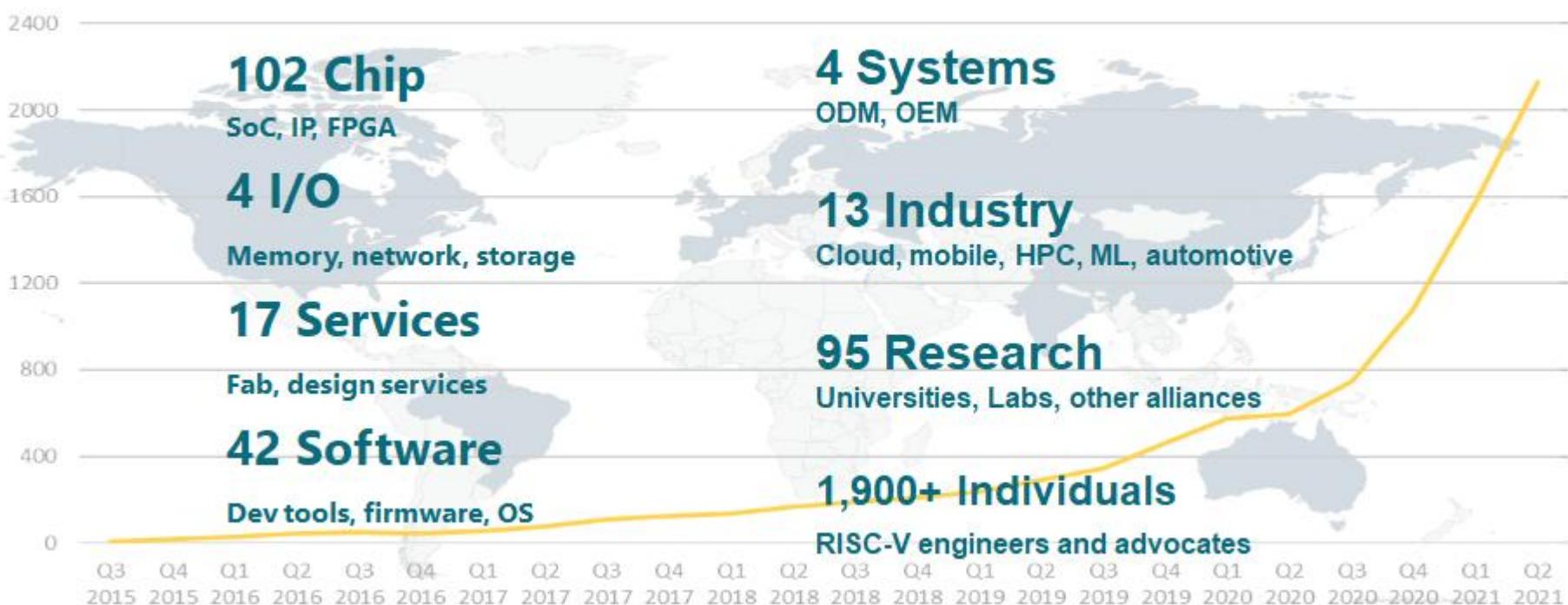
IoT: New growth



Source – Strategy Analytics research services ,October 2017; IoT Strategies , Connected Home Devices, Tablet and Touchscreen Strategies, Wireless Smartphone Strategies, Wearable Device Ecosystem, Smart Home Strategies

RISC-V??

Barriers	Legacy ISA	RISC-V ISA
Complexity	1500+ base instructions Incremental ISA	47 base instructions Modular ISA
Design freedom	\$\$\$ – Limited	Free – Unlimited



Lots of players! (an incomplete list!)

CPU, etc.



AMD

arm

intel

GPGPU,
etc



AMD

intel

TPU,
NPU, etc.

Google

intel

arm

XILINX

cerebras

FPGA,
CPLD, etc.

XILINX

MICROCHIP

AMD

intel

ASIC

openfive

tsmc

SAMSUNG

intel

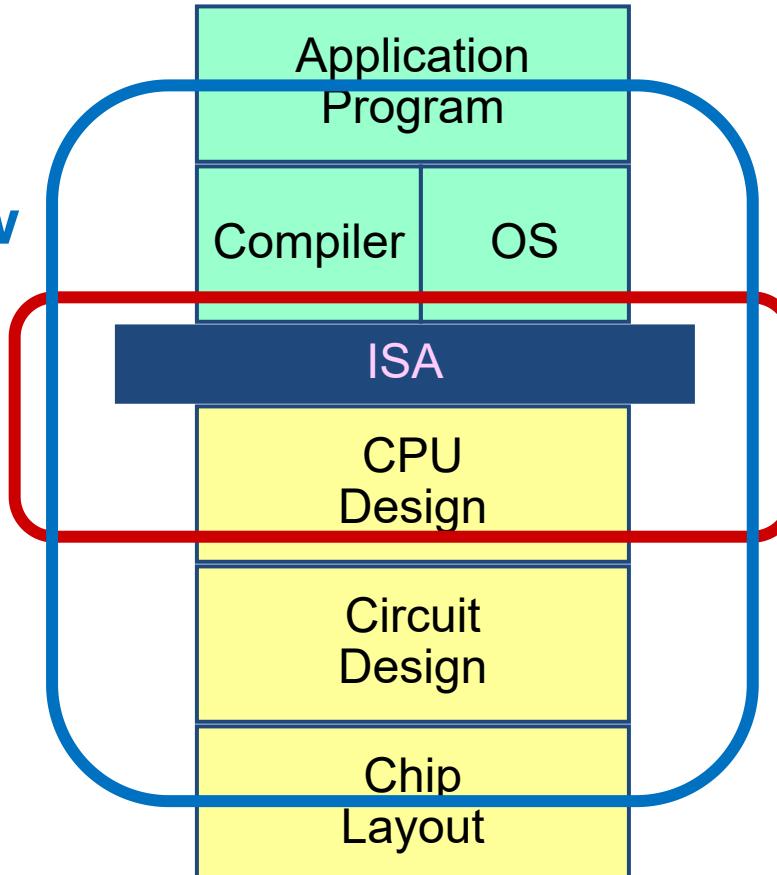
Programmable xPUs

Custom Hardware

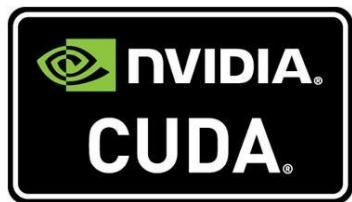
Expand the View of Computer System

Expanded View
of Computer
System

Traditional View
of Computer
System



How do we compile/program for this?



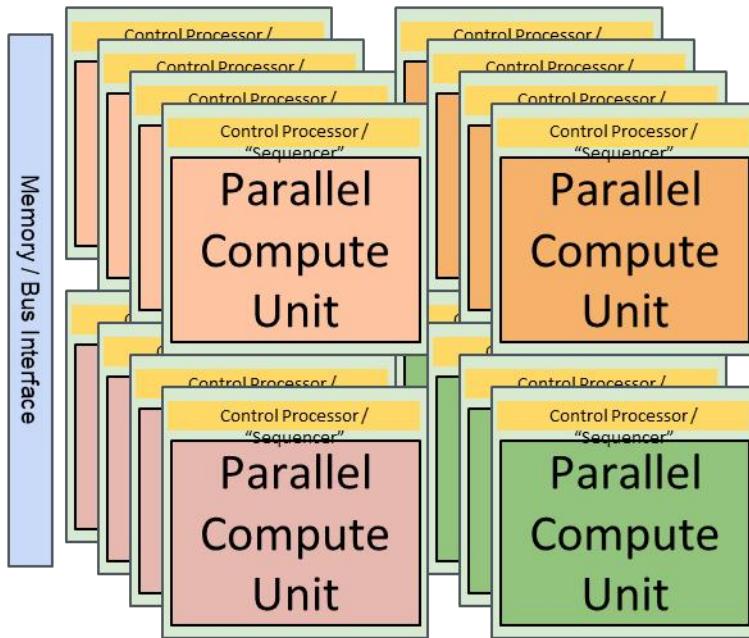
cadence



⇒ Not very compatible, inconsistent quality and scope
... and don't share much code

“DSA Compilers” to the rescue

Hardware



Software

Programming Model + Userspace API

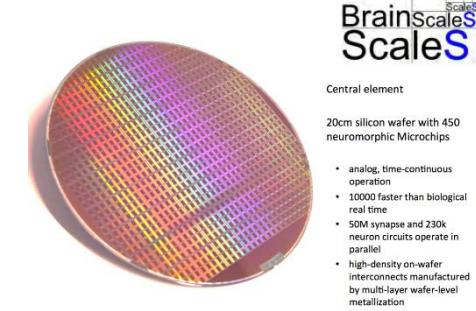
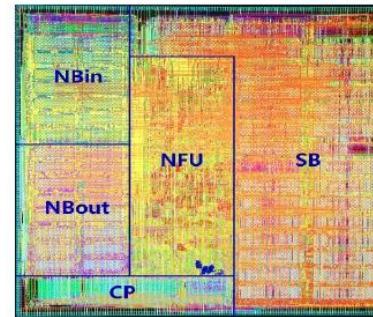
Accelerator Kernel Compiler

Multistream Mgmt / Interop Parallelism
Memory + Communication Optimization
Heterogenous Device + Host fallback
Kernel Code Generation

Control Proc Assembler + Kernel Driver



Figure 3. TPU Printed Circuit Board. It can be inserted into the slot for a SATA disk in a server.



Source: Jouppi et al. “In-Datacenter Performance Analysis of a Tensor Processing Unit.” (2017)

Source: Tianshi Chen, et. al., “DianNao: A Small-Footprint High-Throughput Accelerator for Ubiquitous Machine-Learning” (2014)

Source: Steve Furber et al., “Neuromorphic Computing in the HBP (slides 2017)”

Co-design of HW and SW

- Possible Solution?

CPU. etc.



GPGPU,



TPU,

NPU, etc.
Google



FPGA,



ASIC



Programmable xPUs

Custom Hardware