departamento de informática
FACULDADE DE CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE NOVA DE LISBOA

The Panorama of Parallel and High Performance Computing

Concurrency and Parallelism — 2018-19
Master in Computer Science
(Mestrado Integrado em Eng. Informática)

Joao Lourenço <joao.lourenco@fct.unl.pt>

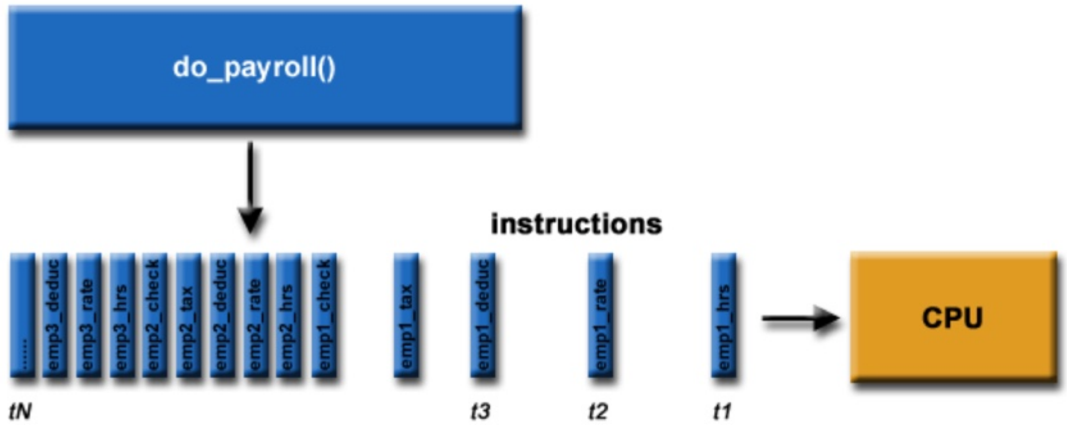
Slides partially based in: https://computing.llnl.gov/tutorials/parallel_comp/

Bibliograpy

- **Chapter 1 of book**
  McCool M., Arch M., Reinders J.;
  Structured Parallel Programming: Patterns for Efficient Computation;
  Morgan Kaufmann (2012);
  ISBN: 978-0-12-415993-8

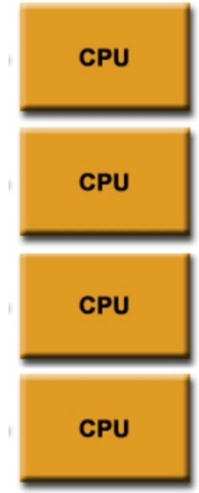
What is Parallel Computing?

- Traditionally, software has been written for serial computation:
  - To be run on a single computer having a single Central Processing Unit (CPU)
  - A problem is broken into a discrete series of instructions
  - Instructions are executed one after another (sequentially)
    - Only one instruction may execute at any moment in time



What is Parallel Computing?

- Is the simultaneous use of multiple compute resources to solve a computational problem:
  - To be executed using multiple processors

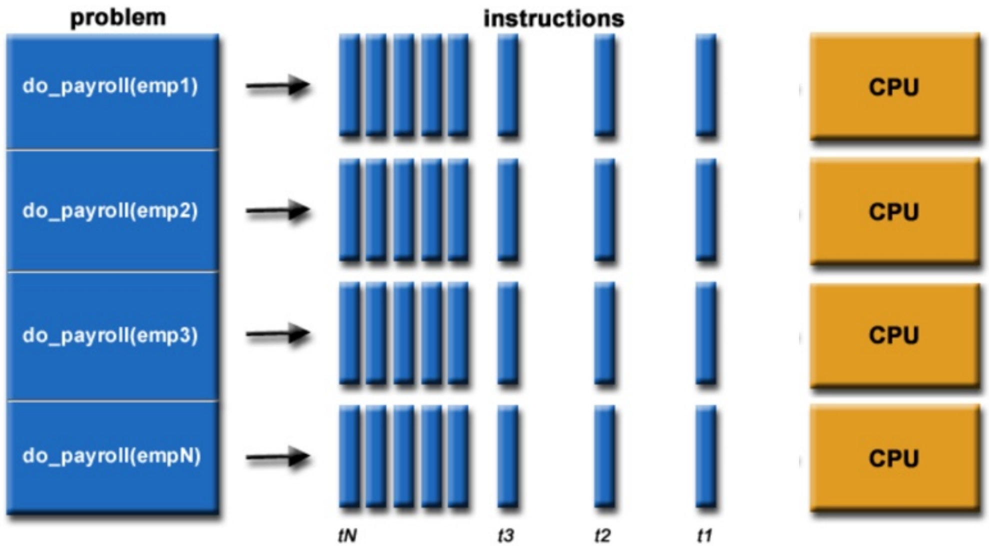


What is Parallel Computing?

- Is the simultaneous use of multiple compute resources to solve a computational problem:
  - To be executed using multiple processors
  - A problem is broken into discrete parts that can be solved concurrently

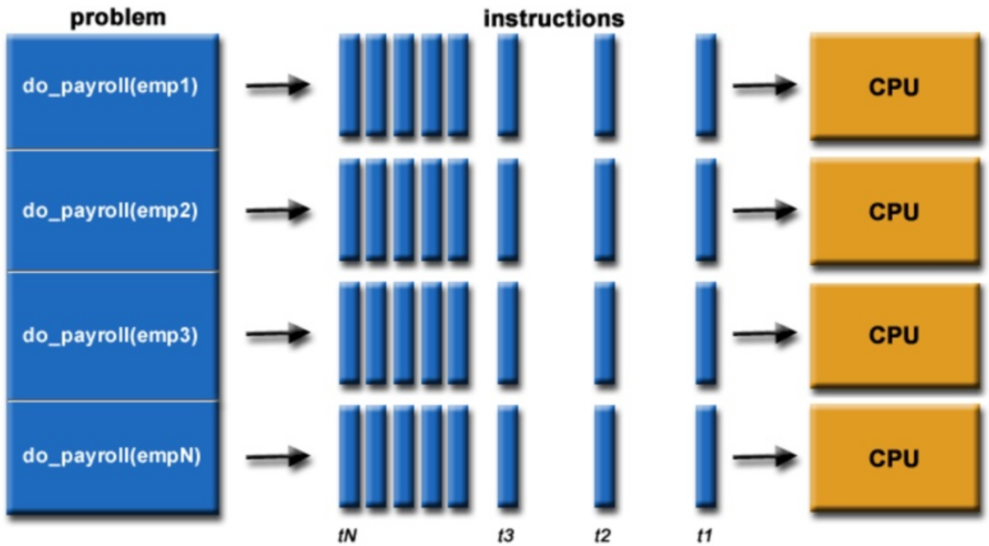
What is Parallel Computing?

- Is the simultaneous use of multiple compute resources to solve a computational problem:
  - To be executed using multiple processors
  - A problem is broken into discrete parts that can be solved concurrently
  - Each part is further broken own to a series of instructions



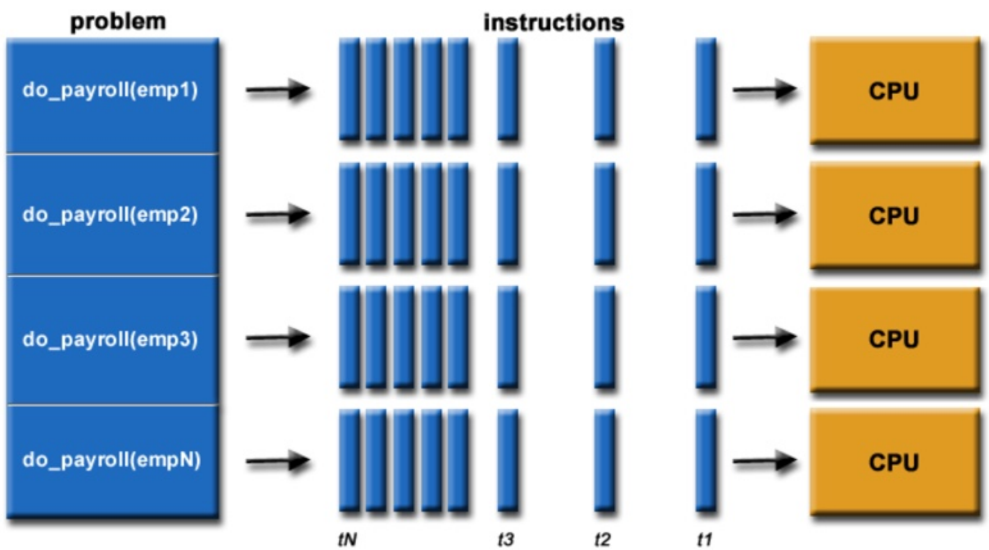
What is Parallel Computing?

- Is the simultaneous use of multiple compute resources to solve a computational problem:
  - To be executed using multiple processors
  - A problem is broken into discrete parts that can be solved concurrently
  - Each part is further broken own to a series of instructions
  - Instructions from each part execute simultaneously on different processors



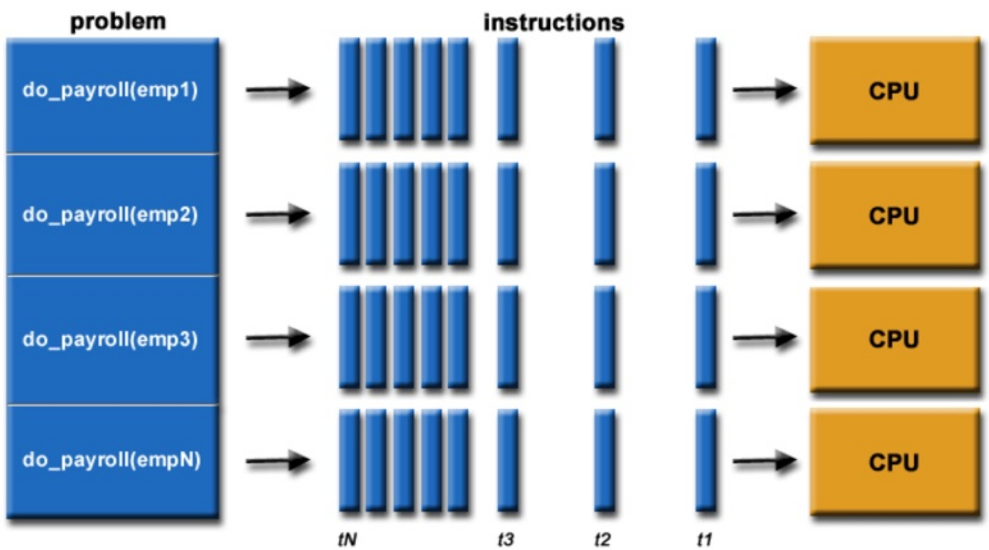
What is Parallel Computing?

- Is the simultaneous use of multiple compute resources to solve a computational problem:
  - To be executed using multiple processors
  - A problem is broken into discrete parts that can be solved concurrently
  - Each part is further broken own to a series of instructions
  - Instructions from each part execute simultaneously on different processors
  - An overall control/coordination mechanism is employed



What is Parallel Computing?

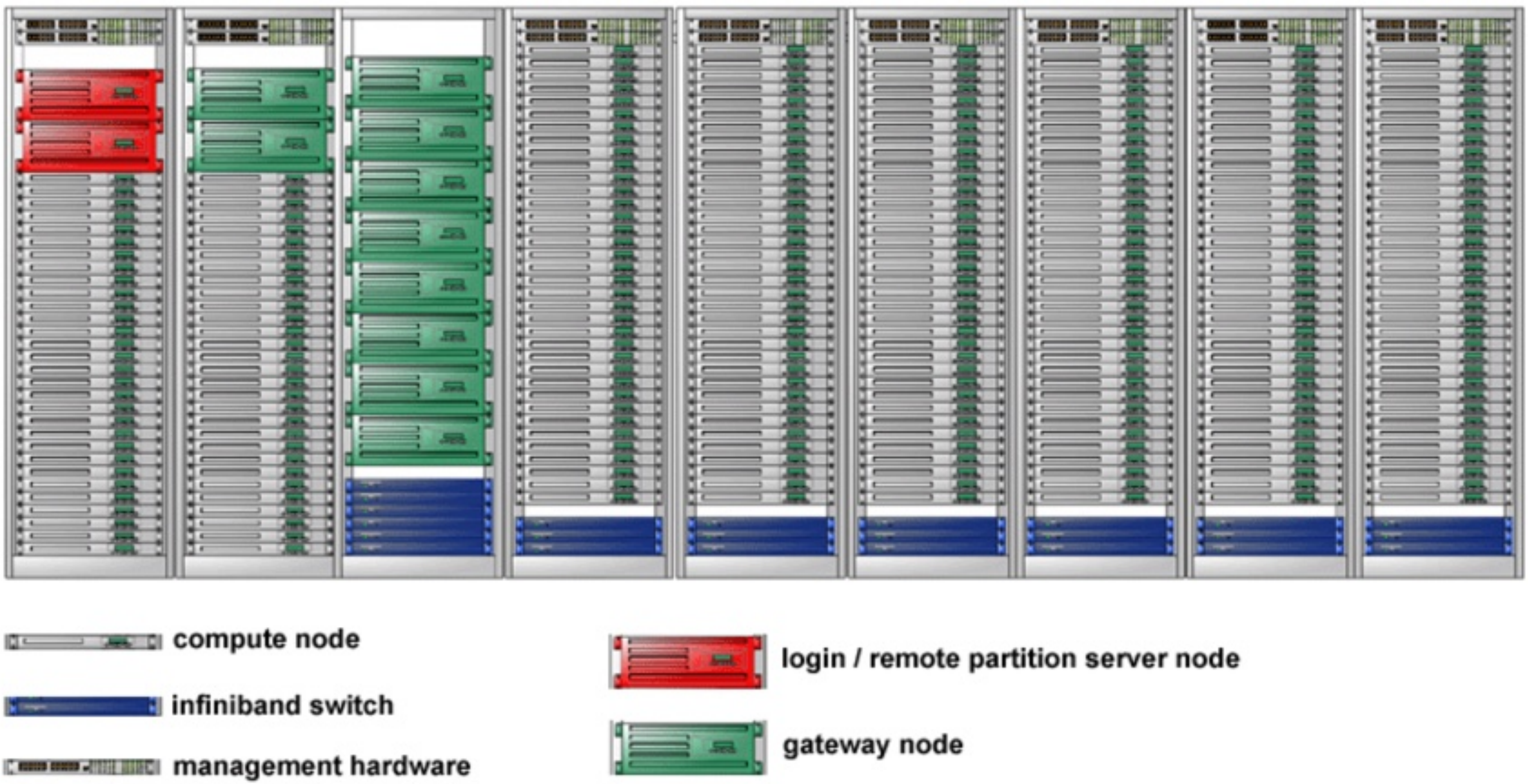
- Is the simultaneous use of multiple compute resources to solve a computational problem:
  - To be executed using multiple processors
  - A problem is broken into discrete parts that can be solved concurrently
  - Each part is further broken own to a series of instructions
  - Instructions from each part execute simultaneously on different processors
  - An overall control/coordination mechanism is employed



What is Parallel Computing?

- The computational problem should be able to:
  - Be broken apart into discrete pieces of work that can be solved simultaneously
  - Execute multiple program instructions at any moment in time
  - Be solved in less time with multiple compute resources than with a single compute resource
- The computing resources might be:
  - A single computer with multiple processors
  - An arbitrary number of computers connected by a network (real or virtual systems)
  - A combination of both

What is Parallel Computing?



Sunway TaihuLight System

Summit - IBM Power System AC922

Units of Measures

- High Performance Computing (HPC) units are:
  - Flop: floating point operation, usually double precision unless noted - Flop/s: floating point operations per second
  - Bytes: size of data (a double precision floating point number is 8)
- Typical sizes are millions, billions, trillions…

Mega	Mbyte = 2^20 = 1048576 ~ 10^6 bytes	Mflop/s = 10^6 flop/s
Giga	Gbyte = 2^30 ~ 10^9 bytes	Gflop/s = 10^9 flop/s
Tera	Tbyte = 2^40 ~ 10^12 bytes	Tflop/s = 10^12 flop/s
Peta	Pbyte = 2^50 ~ 10^15 bytes	Pflop/s = 10^15 flop/s
Exa	Ebyte = 2^60 ~ 10^18 bytes	Eflop/s = 10^18 flop/s
Zetta	Zbyte = 2^70 ~ 10^21 bytes	Zflop/s = 10^21 flop/s
Yotta	Ybyte = 2^80 ~ 10^24 bytes	Yflop/s = 10^24 flop/s

Top500.org — June 2018

Rank	System	Cores	Rmax (TFlop/s)	Rpeak (TFlop/s)	Power (kW)
1	**Summit** - IBM Power System AC922, IBM POWER9 22C 3.07GHz, NVIDIA Volta GV100, Dual-rail Mellanox EDR Infiniband (/system/179397), IBM DOE/SC/Oak Ridge National Laboratory (/site/48553) United States	2,282,544	122,300.0	187,659.3	8,806
2	**Sunway TaihuLight** - Sunway MPP, Sunway SW26010 260C 1.45GHz, Sunway (/system/178764), NRCPC National Supercomputing Center in Wuxi (/site/50623) China	10,649,600	93,014.6	125,435.9	15,371
3	**Sierra** - IBM Power System S922LC, IBM POWER9 22C 3.1GHz, NVIDIA Volta GV100, Dual-rail Mellanox EDR Infiniband (/system/179398), IBM DOE/NNSA/LLNL (/site/49763) United States	1,572,480	71,610.0	119,193.6	
4	**Tianhe-2A** - TH-IVB-FEP Cluster, Intel Xeon E5-2692v2 12C 2.2GHz, TH Express-2, Matrix-2000 (/system/177999), NUDT National Super Computer Center in Guangzhou (/site/50365) China	4,981,760	61,444.5	100,678.7	18,482
5	**AI Bridging Cloud Infrastructure (ABCI)** - PRIMERGY CX2550 M4, Xeon Gold 6148 20C 2.4GHz, NVIDIA Tesla V100 SXM2, Infiniband EDR (/system/179393), Fujitsu National Institute of Advanced Industrial Science and Technology (AIST) (/site/50762) Japan	391,680	19,880.0	32,576.6	1,649

1 TFLOP in a chip

Intel Core i9 X-series

Cost $2 000

Rank	System	Cores	Rmax (TFlop/s)	Rpeak (TFlop/s)	Power (kW)
1	**Summit** - IBM Power System AC922, IBM POWER9 22C 3.07GHz, NVIDIA Volta GV100, Dual-rail Mellanox EDR Infiniband (/system/179397), IBM DOE/SC/Oak Ridge National Laboratory (/site/48553) United States	2,282,544	122,300.0	187,659.3	8,806

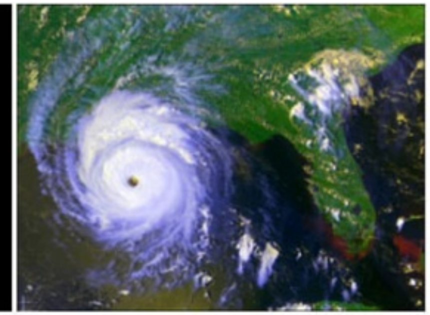
The Real World is Massively Parallel

- In the natural world, many complex, interrelated events are happening at the same time, yet within a temporal sequence.
- Compared to serial computing, parallel computing is much better suited for modeling, simulating and understanding complex, real world phenomena.
- For example, imagine modeling serially the following systems.

The Real World is Massively Parallel


Galaxy Formation


Planetary Movments


Climate Change

Rush Hour Traffic


Plate Tectonics


Weather

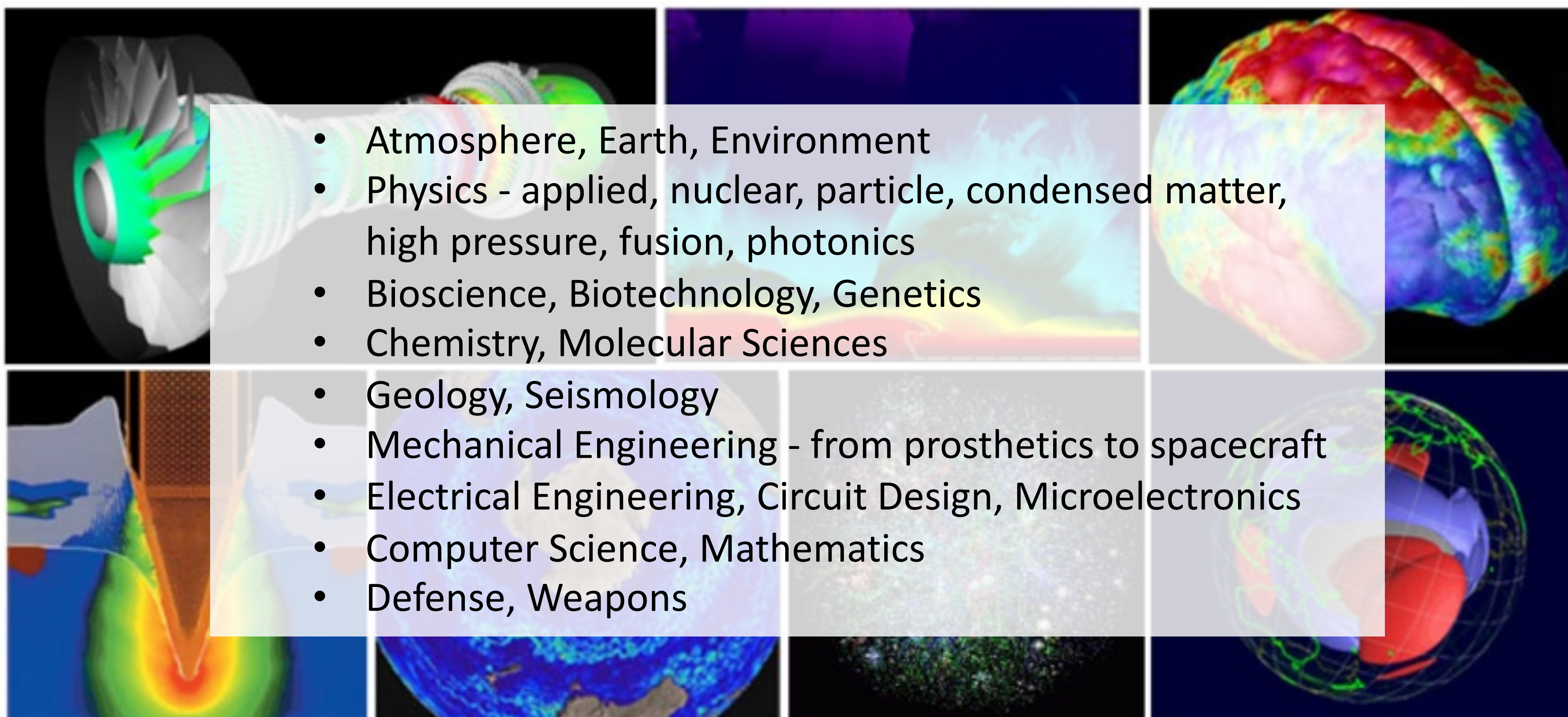

Auto Assembly


Jet Construction


Drive-thru Lunch

Uses for Parallel Computing

- Modeling difficult problems in many areas of science and engineering



- Atmosphere, Earth, Environment
- Physics - applied, nuclear, particle, condensed matter, high pressure, fusion, photonics
- Bioscience, Biotechnology, Genetics
- Chemistry, Molecular Sciences
- Geology, Seismology
- Mechanical Engineering - from prosthetics to spacecraft
- Electrical Engineering, Circuit Design, Microelectronics
- Computer Science, Mathematics
- Defense, Weapons

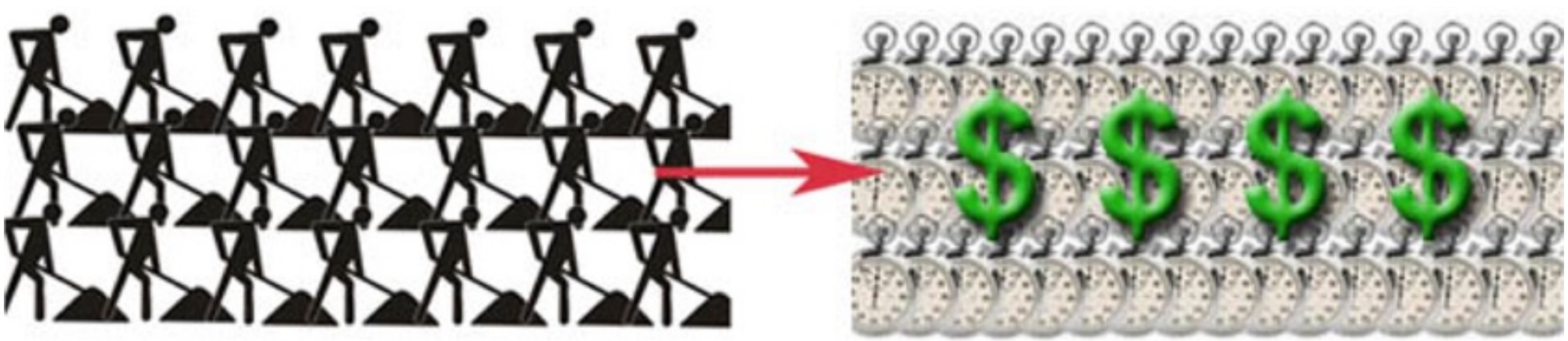
Uses for Parallel Computing

- Industrial and Commercial

- Databases, data mining
- Oil exploration
- Web search engines, web based business services
- Medical imaging and diagnosis
- Pharmaceutical design
- Financial and economic modeling
- Management of national and multi-national corporations
- Advanced graphics and virtual reality, particularly in the entertainment industry
- Networked video and multi-media technologies
- Collaborative work environments

Why Use Parallel Computing?

- Save time and/or money
  - In theory, throwing more resources at a task will shorten its time to completion, with potential cost savings. Parallel computers can be built from cheap, commodity components.

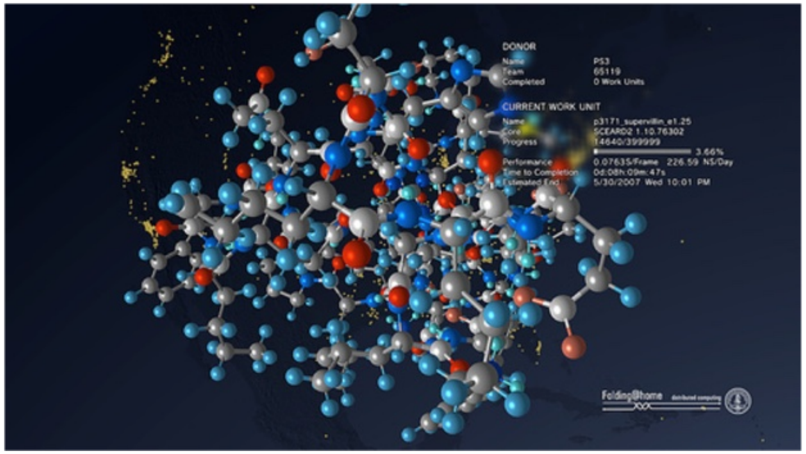


Why Use Parallel Computing?

- Solve larger problems
  - Many problems are so large and/or complex that it is impractical or impossible to solve them on a single computer. For example:
    - "Grand Challenge" (en.wikipedia.org/wiki/Grand_Challenge) problems requiring PetaFLOPS and PetaBytes of computing resources.
    - Web search engines/databases processing millions of transactions per second.

Why Use Parallel Computing?

- Use of non-local resources
  - Using compute resources on a wide area network, or even the Internet when local compute resources are scarce. For example:
    - SETI@home (setiathome.berkeley.edu) over 1.6 million users, 4 million computers, in nearly every country in the world. Source: https://setiathome.berkeley.edu/stats.php (Sep, 2016).
    - Folding@home (folding.stanford.edu) uses over 320,000 computers globally (Sep, 2016)



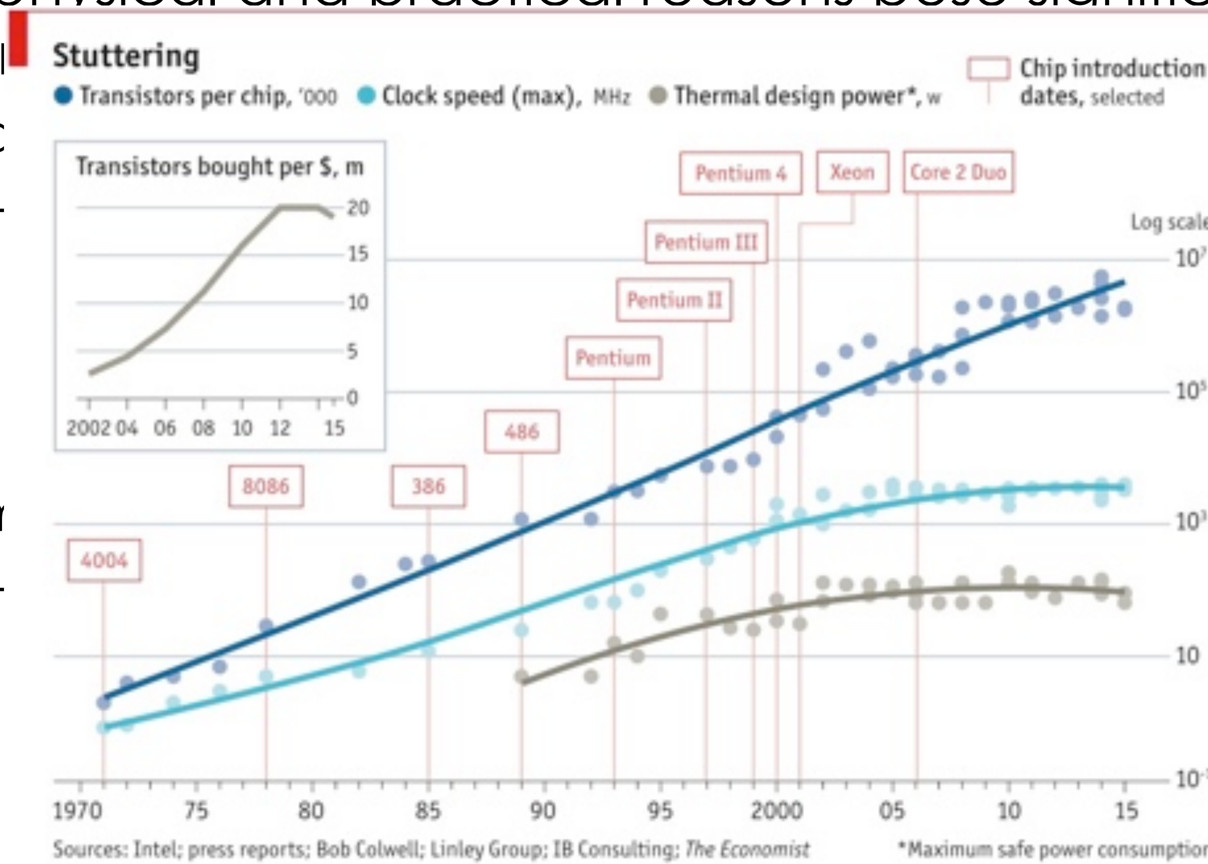
Limits to serial computing

- Both physical and practical reasons pose significant constraints to simply building ever faster serial computers:
  - Transmission speeds
    - the speed of a serial computer is directly dependent upon how fast data can move through hardware. Absolute limits are the speed of light (30 cm/nanosecond) and the transmission limit of copper wire (9 cm/nanosecond). Increasing speeds requires increasing proximity of processing elements.
  - Limits to miniaturization
    - processor technology is allowing an increasing number of transistors to be placed on a chip. However, even with molecular or atomic-level components, a limit will be reached on how small components can be.

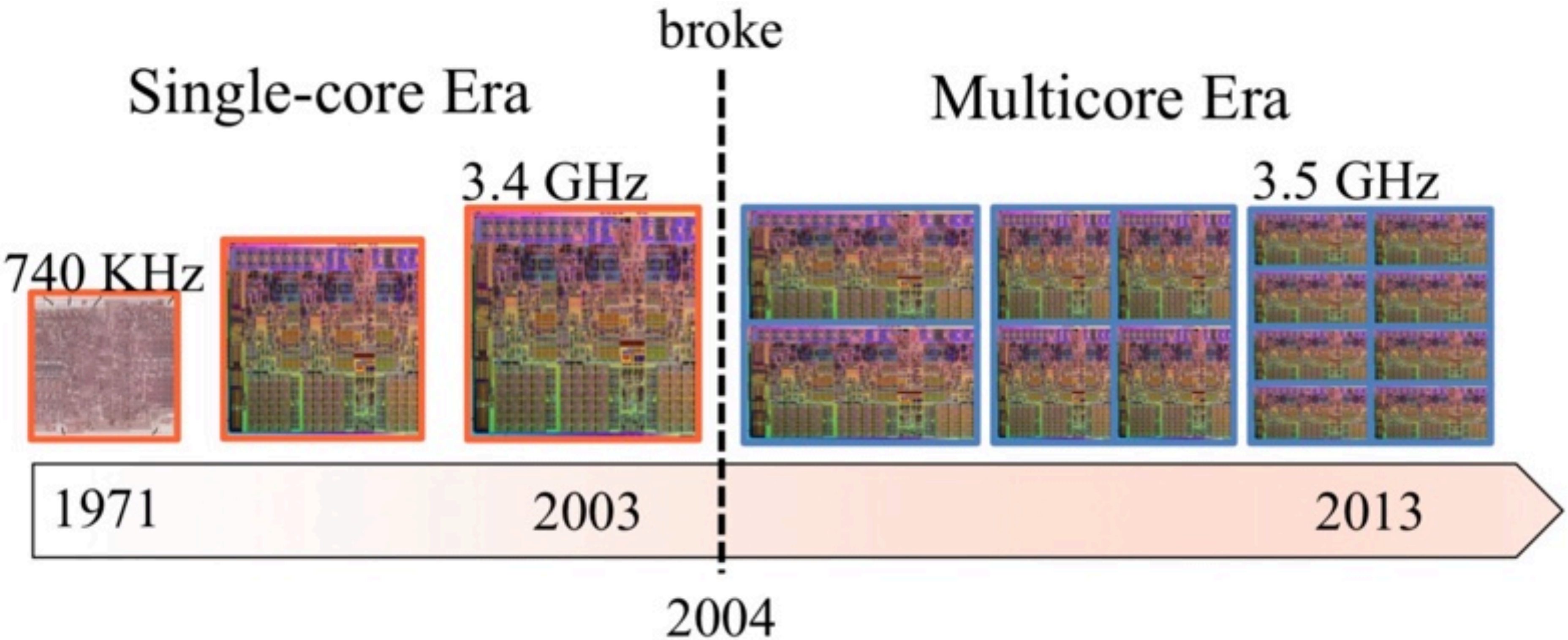
Limits to serial computing

– Both physical and practical reasons pose significant constr... mputers:

- Trc...
  - ... nt upon how ... nits are the ... ission limit of ... ls ... ents.
- Lin...
  - ... ber of ... with ... e reached



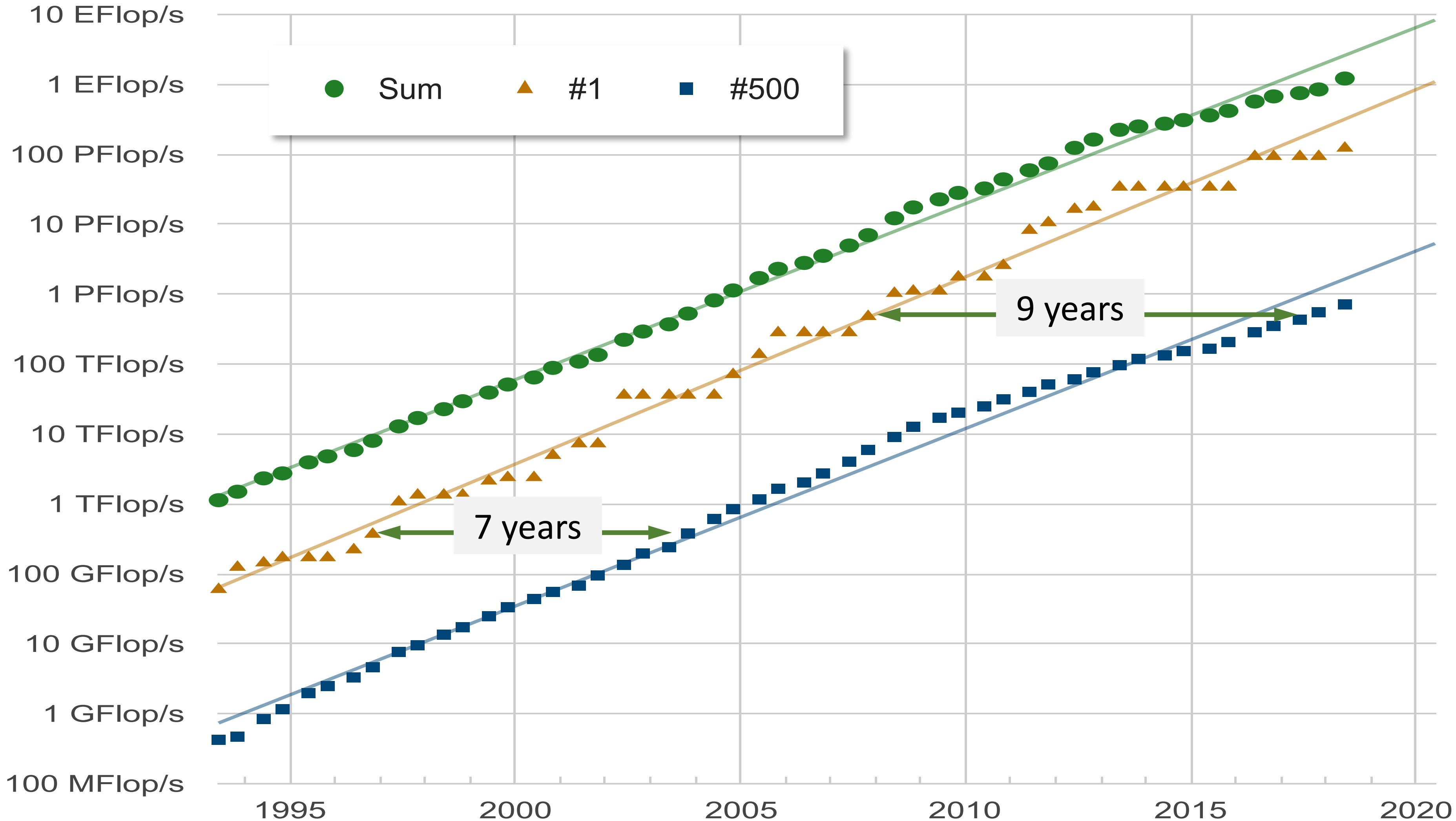
Limits to serial computing



The Future

- During the past 20+ years, the trends indicated by ever faster networks, distributed systems, and multi-processor computer architectures (even at the desktop level) clearly show that parallelism is the future of computing.
- In this same time period, there has been a greater than 1000x increase in supercomputer performance, with no end currently in sight.
- The race is already on for Exascale Computing! (10^{18} FLOPS)

Performance development



The END