

TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN TOÁN ỨNG DỤNG & TIN HỌC



BÀI GIẢNG
TÍNH TOÁN SONG SONG

Giảng viên: Đoàn Duy Trung
Bộ môn : Toán Tin



BÀI 1. GIỚI THIỆU CHUNG



NỘI DUNG BÀI HỌC

- Sơ lược về lịch sử hình thành và phát triển
- Tổng quan về tính toán song song



0. SƠ LƯỢC VỀ LỊCH SỬ HÌNH THÀNH KHÁI NIỆM TÍNH TOÁN SONG SONG

0.1.Sơ lược về lịch sử

- 1955: IBM giới thiệu kiến trúc 704. Principal với tên gọi là Gene Amdahl, Máy tính thương mại đầu tiên trên thế giới với floating-point – khả năng thực hiện 5kFLOPS.
- 1956:
 - IBM bắt đầu với dự án STRETCH giới thiệu siêu máy tính cho LANL (Los Alamos National Laboratory). Sản xuất ra máy tính gấp 100 lần năng lực máy tính cùng thời điểm.
 - LARC (Livermore Automatic Research Computer) siêu máy tính cho Lawrence Livermore National Laboratory

0.1.Sơ lược về lịch sử (tt)

- 1958:
 - Tại Pháp xuất hiện Gamma 60 với multiple functional units và toán tử fork&joins
 - John Cocke và Daniel Slotnick thảo luận về sử dụng song song trong các phép toán số học tại phòng thí nghiệm của IBM
- 1959:
 - Máy tính STRETCH đầu tiên ra đời (8 cái)
 - Máy tính LARC ra đời (2 cái)

0.1.Sơ lược về lịch sử (tt)

- 1960:
 - Control Data bắt đầu phát triển CDC6600
 - Honeywell 800 với phần cứng hỗ trợ chia sẻ 8 chương trình
 - E.V.Yeveinov tại Novosibirsk bắt đầu kiến trúc song song
- 1962:
 - CDC 1604 đầu tiên xuất hiện
 - Atlas Computer – máy tính đầu tiên sử dụng Virtual Memory and paging. – 200kFLOPS

0.2. Giới thiệu về chuẩn LINPACK

- Giới thiệu:
 - Prof. Jack Dongarra, Jim Bunch, Cleve Moler và Gilbert Stewart phát triển trên ngôn ngữ Fortran.
 - Thư viện phần mềm dùng để đo khả năng tính toán đại số tuyến tính.
 - Sử dụng cho các siêu máy tính vào năm 1970 và 1980.
 - Sử dụng thư viện BLAS (Basic Linear Algebra Subprograms) thực hiện các phép toán vectơ và ma trận
 - HPL (High Performance Linpack) – xếp hạng máy tính trong top500.org

0.3. Giới thiệu về siêu máy tính

- Đơn vị đo:
 - Flops – Floating point Operation Per Second:
Số phép toán thực hiện trên một giây.
 - Các đơn vị khác:
 - 1 MFlop/s = 10^6 flops
 - 1 GFlop/s = 10^9 flops
 - 1 TFlop/s = 10^{12} flops
 - 1 PFlop/s = 10^{15} flops
- Xếp hạng siêu máy tính đưa ra bởi hội thảo International Supercomputer (6 tháng / lần).
Thông tin tại top500.org

0.3. Giới thiệu về siêu máy tính (tt)

- Độ đo lý thuyết và cực đại:
 - Rpeak: là độ đo lý thuyết tối đa của siêu máy tính do nhà sản xuất công bố
 - Xác định bằng số phép tính cộng và nhân được hoàn thành trong suốt 1 chu kỳ thời gian của máy tính.
 - VD: Máy tính 2GHz hoàn thành 4 phép tính trong 1 cycle thì hiệu suất lý thuyết là 8GFlop/s
 - Rmax: Là độ đo thực tế mà siêu máy tính có thể thực hiện được do phần mềm tính toán (LINPACK)

0.3. Giới thiệu về siêu máy tính

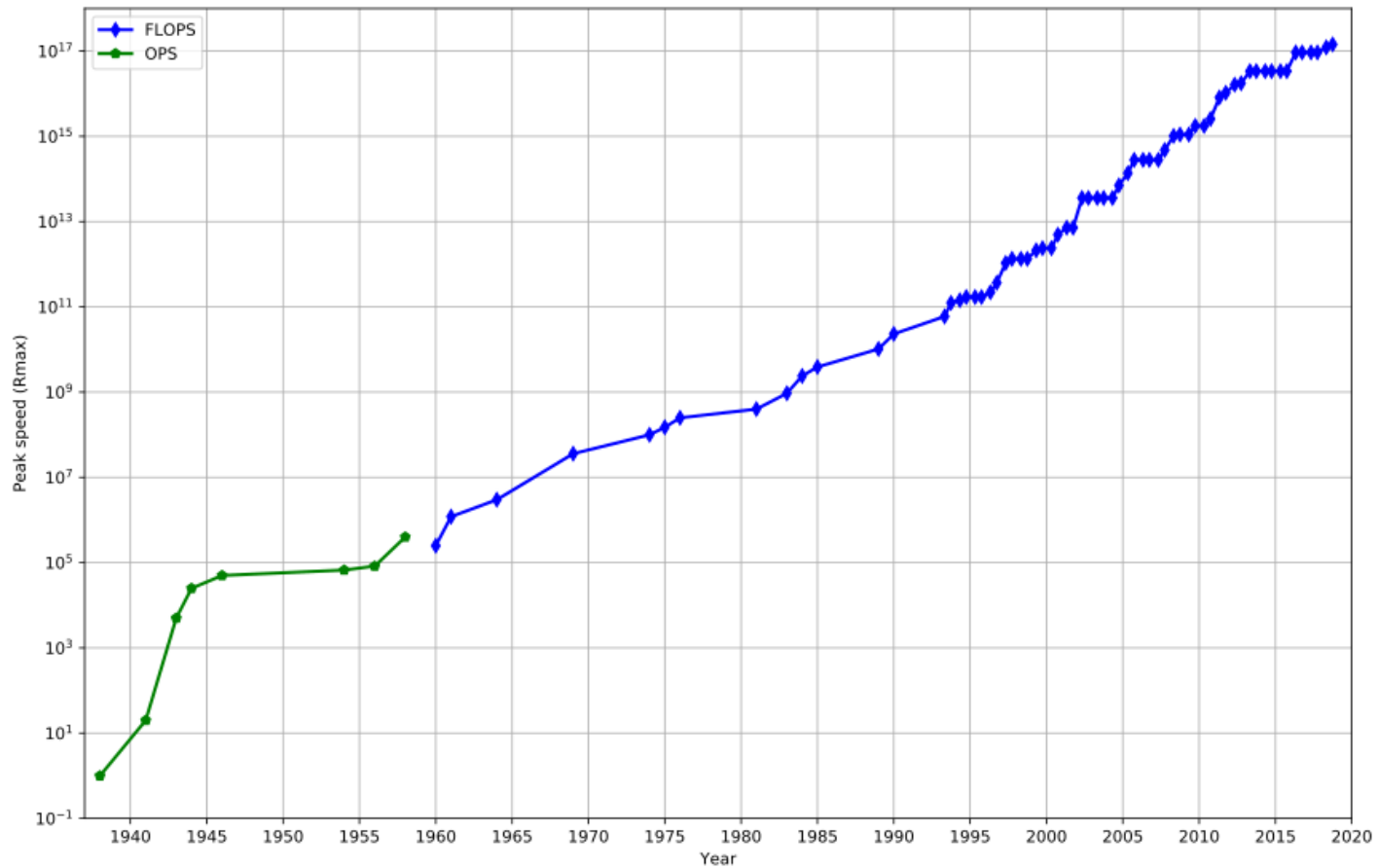
- Xếp hạng vào 6/2005:
 - Vị trí số 1 là BlueGene/L System (No1 vào 11/2004)
 - Tăng gấp đôi sau 6 tháng và thời điểm 6/2005 đạt chỉ số Linpack là 136.8 TFlop/s (trillions of calculations per second)

0.3. Giới thiệu về siêu máy tính (tt)

■ Siêu máy tính:

- Siêu máy tính là những máy tính có khả năng tính toán rất lớn;
- Hiệu suất của siêu máy tính được đo bằng đơn vị FLOPS thay vì MIPS (Million instructions per second)
- Khả năng tính toán phụ thuộc nhiều vào tốc độ của CPU (bộ vi xử lý) -> phụ thuộc vào cấu trúc và số lượng transistors chứa trong CPU
- Định luật Moore (người đồng sáng lập ra Intel): Số transistors tích hợp trên CPU tăng gấp đôi sau chu kỳ 18 (24) tháng.

Top supercomputer speeds

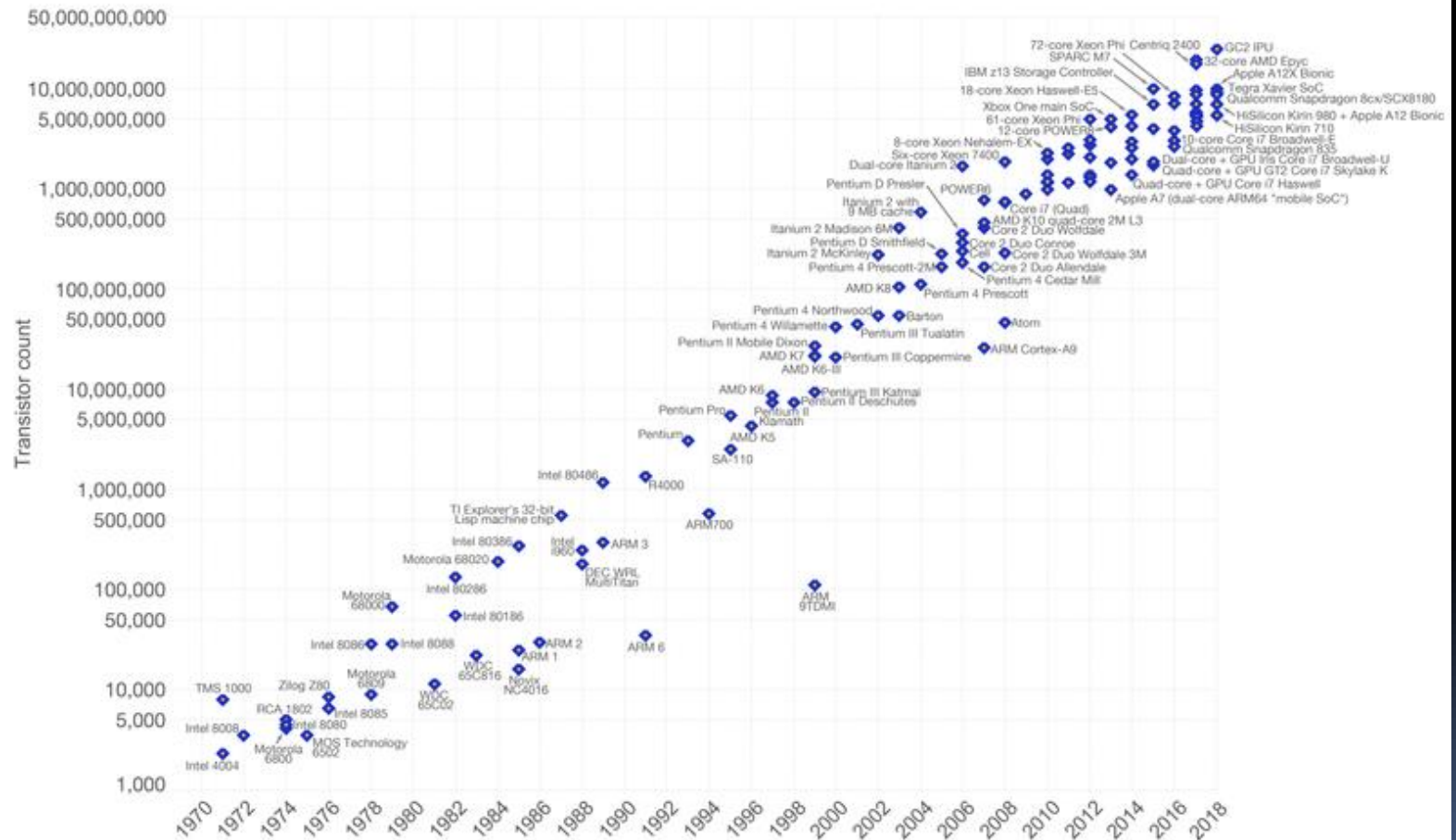


0.3. Giới thiệu về siêu máy tính (tt)

Moore's Law – The number of transistors on integrated circuit chips (1971-2018)

Moore's law describes the empirical regularity that the number of transistors on integrated circuits doubles approximately every two years. This advancement is important as other aspects of technological progress – such as processing speed or the price of electronic products – are linked to Moore's law.

Our World
in Data



Data source: Wikipedia (https://en.wikipedia.org/wiki/Transistor_count)

The data visualization is available at OurWorldInData.org. There you find more visualizations and research on this topic.

Licensed under CC-BY-SA by the author Max Roser.

0.3. Giới thiệu về siêu máy tính (tt)

■ Tháng 11 /2019

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 , IBM DOE/SC/Oak Ridge National Laboratory United States	2,414,592	148,600.0	200,794.9	10,096
2	Sierra - IBM Power System AC922, IBM POWER9 22C 3.1GHz, NVIDIA Volta GV100, Dual-rail Mellanox EDR Infiniband , IBM / NVIDIA / Mellanox DOE/NNSA/LLNL United States	1,572,480	94,640.0	125,712.0	7,438
3	Sunway TaihuLight - Sunway MPP, Sunway SW26010 260C 1.45GHz, Sunway , NRCPC National Supercomputing Center in Wuxi China	10,649,600	93,014.6	125,435.9	15,371
4	Tianhe-2A - TH-IVB-FEP Cluster, Intel Xeon E5-2692v2 12C 2.2GHz, TH Express-2, Matrix-2000 , NUDT National Super Computer Center in Guangzhou China	4,981,760	61,444.5	100,678.7	18,482
5	Frontera - Dell C6420, Xeon Platinum 8280 28C 2.7GHz, Mellanox InfiniBand HDR , Dell EMC Texas Advanced Computing Center/Univ. of Texas United States	448,448	23,516.4	38,745.9	

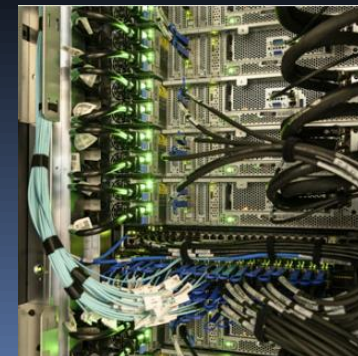
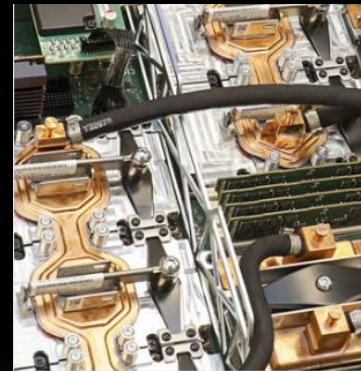
0.3. Giới thiệu về siêu máy tính

- Tháng 11/2019:
 - Summit là siêu máy tính phát triển bởi IMB sử dụng tại Oak Ridge National Laboratory.
 - Siêu máy tính mạnh mẽ nhất thế giới với tốc độ xử lý 200 petaFLOPS.
 - Top 3 máy tính tiết kiệm điện năng
 - Summit là siêu máy tính đầu tiên trên thế giới đạt đến tốc độ exaop (10^{12} phép tính trên giây)



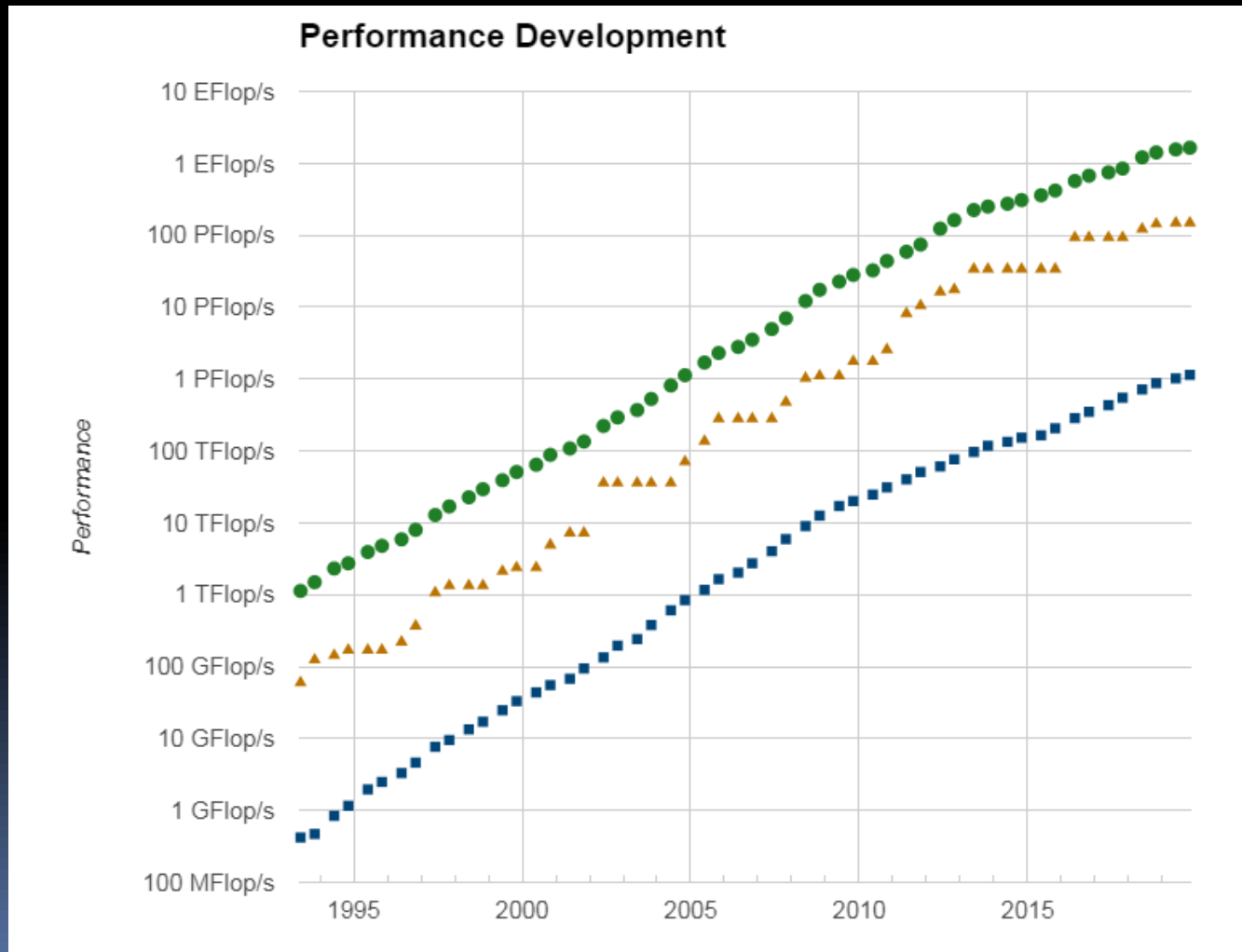
Summit – Siêu máy tính

- Một vài thông số:
 - 200 quadrillion calculations per second (10^{15})
 - Bộ nhớ 250 pbytes
 - 9216 IBM Power9 CPUs
 - Tốc độ truyền giữa các nút: 25 gigabytes/s
 - 27648 NVIDIA Tesla GPUs



0.3. Giới thiệu về siêu máy tính (tt)

- Performance Development



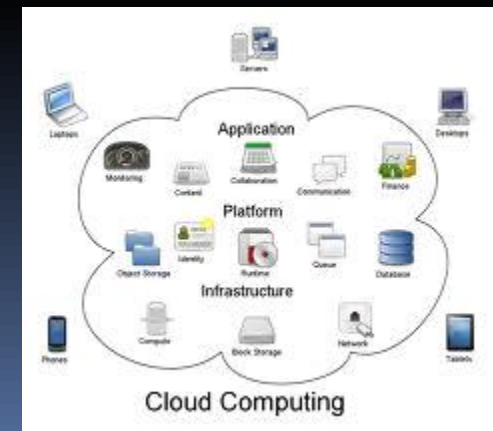
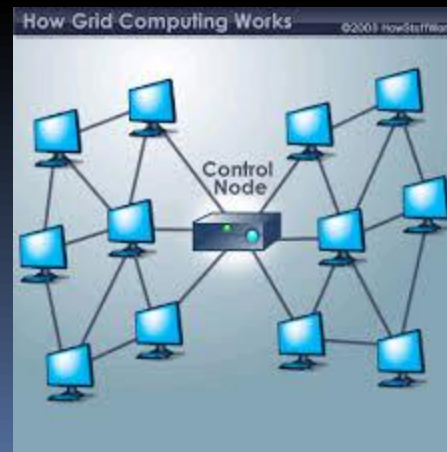
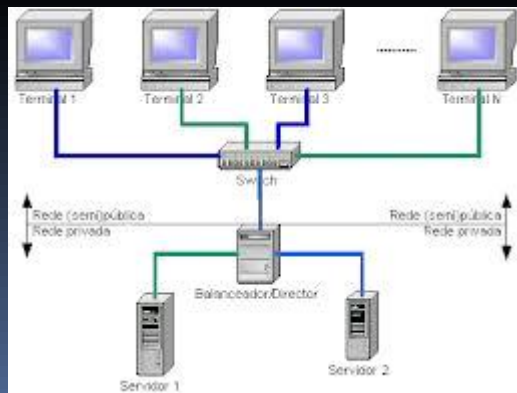
0.4. Xu hướng phát triển của máy tính

- 2 xu hướng đang được các nhà khoa học đẩy mạnh phát triển các máy tính hỗ trợ các tính toán khoa học:
 - Hiệu suất dựa trên hệ thống máy
 - Hiệu suất dựa trên 1 máy đơn lẻ



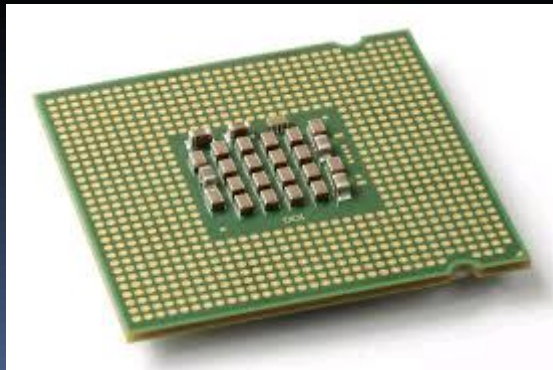
0.4. Xu hướng phát triển của máy tính

- Hiệu suất trên hệ thống máy:
 - Cluster Computing
 - P2P Computing
 - Grid Computing
 - Cloud Computing
 -



0.4. Xu hướng phát triển của máy tính

- Hiệu suất trên 1 máy đơn:
 - 2 xu thế:
 - Có nhiều bộ vi xử lý , nhưng không tốn nhiều năng lượng
 - Tận dụng nguồn tài nguyên đã có để tăng khả năng tính toán.



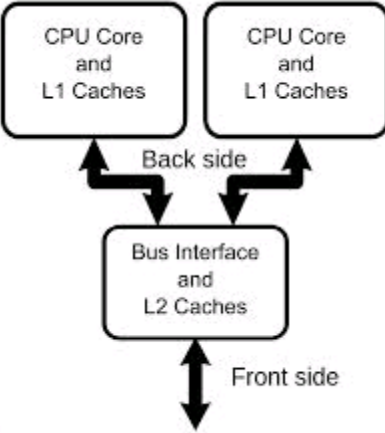
0.4. Xu hướng phát triển của máy tính

- Khả năng tính toán của máy tính vẫn ngày 1 cải thiện mặc dù:
 - Nguồn năng lượng không tăng đáng kể
 - Tỏa nhiệt không nhiều
 - Kích thước giảm...



SECRET

-



0.4. Xu hướng phát triển của máy tính

■ Multi-Cores:

- 4/2005: AMD giới thiệu Multi-Core Opterons đầu tiên – sử dụng dual-core với 2 processor cores
- Thế hệ thứ 2 Opterons bao gồm 2 loại: single socket, dual socket-capable, quad or octo socket capable
- Thế hệ thứ 3 quad-core (4 cores) ra đời vào 9/2007
- Thế hệ thứ 4 ra đời vào tháng 6/2009: bộ vi xử lý 6 cores tên là Istanbul với tốc độ tăng 34% so với 4 core.



0.4. Xu hướng phát triển của máy tính

- Multi-Cores:

- Dual-core processor: 2 core (AMD Phenom X2, Intel Core Duo)
- Quad-core processor: 4 core (AMD Phenom X4, Intel 2010)
- Hexa-core processor: 6 core (AMD Phenom X6, Intel Core i7 Extreme Edition 980X)
- Octa-core processor: 8 core (AMD FX-8150)

0.4. Xu hướng phát triển của máy tính

- Intel Hyper-Threading:
 - Tăng khả năng sử dụng nguồn tài nguyên processor, cho phép đa luồng (threads) chạy trên mỗi core, nâng cao khả năng xử lý tổng thể .
 - Cho phép chạy đồng thời nhiều ứng dụng đòi hỏi khả năng xử lý cao
 - Giữ cho hệ thống được bảo vệ tốt, hoạt động hiệu quả

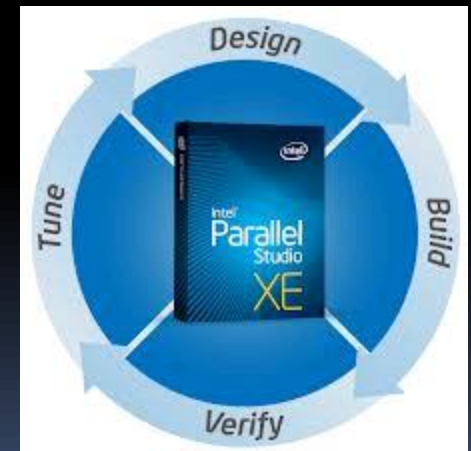
0.5. Phần mềm & Lập trình

- Xu thế chung hiện nay:



0.5. Phần mềm & Lập trình

- Phần mềm:
 - Hiện nay các máy tính hầu hết đều là multiple core (tối thiểu cũng là Dual Core)
 - Số lượng phần mềm khai thác tối đa khả năng hoạt động trên chưa nhiều (Cả hệ điều hành cũng ít hỗ trợ)
 - Intel Parallel Studio tích hợp vào Microsoft Visual Studio.



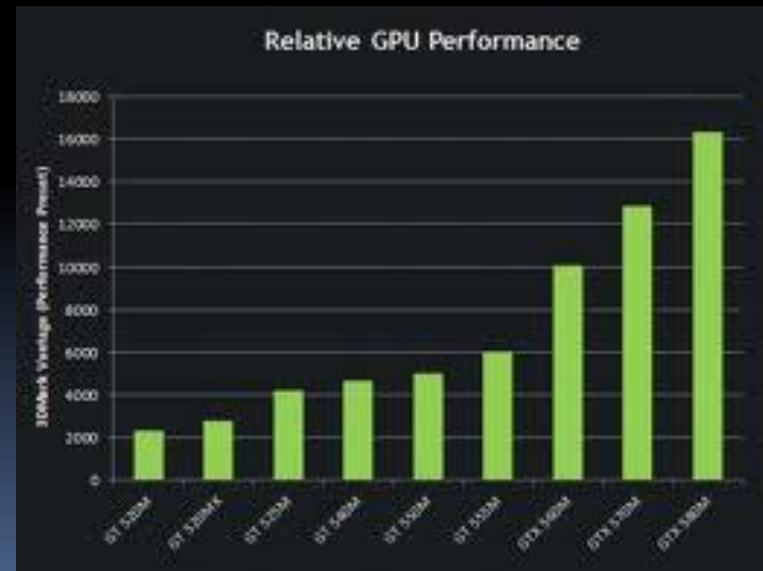
0.5. Phần mềm & Lập trình

- OpenCL:

- Tạo ra bởi Khronos Group là 1 framework cho phép thực hiện chương trình qua CPUs, GPUs,...
- Cung cấp lập trình song song trên task-based và data-based
- Hãng Apple từ 9/2009 cung cấp hệ điều hành Snow Leopard tích hợp OpenCL (Open Computing Language)
- Mac OS Lion, OpenCL được hỗ trợ nhiều hơn
- Sử dụng IDE Xcode để lập trình

0.5. Phần mềm & Lập trình

- GPU:
 - Từ năm 2007 Nvidia sử dụng chip đồ họa như bộ xử lý trung tâm (CPU)



0.5. Phần mềm & Lập trình

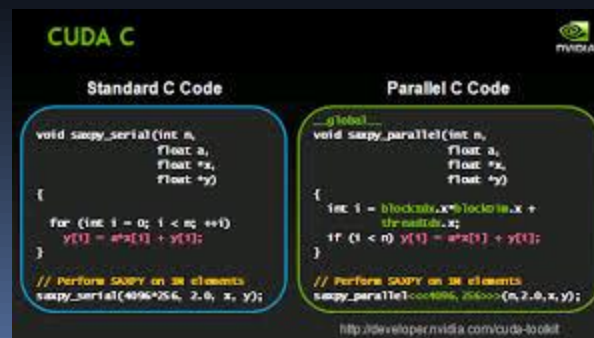
- CPU kết hợp GPU:
 - Khi đó máy tính bao gồm 2 bộ xử lý tồn tại song song
 - GPU – Graphics Process Unit: bộ xử lý đồ họa hoạt động song song cùng CPU

The Difference between a CPU and GPU



0.5. Phần mềm & Lập trình

- Ngôn ngữ lập trình GPU:
 - Sử dụng ngôn ngữ lập trình nào để khai thác khả năng chạy trên GPU??
 - NVIDIA đề xuất và phát triển ngôn ngữ CUDA (Compute Unified Device Architecture) cho phép các lập trình viên sử dụng khả năng GPU để tính toán tương tự như sử dụng CPU



0.5. Phần mềm và lập trình

- CUDA:
 - Là mô hình lập trình song song
 - Hỗ trợ nhiều hệ điều hành khác nhau như Windows, Linux, MacOS

The screenshot displays the NVIDIA Developer Zone website. At the top, the NVIDIA logo and 'DEVELOPER ZONE' text are visible, along with a 'Log In' link. A navigation bar includes links for DEVELOPER CENTERS, TECHNOLOGIES, TOOLS, RESOURCES, and COMMUNITY. A search bar labeled 'Search DevZone' is also present. The main section is titled 'CUDA ZONE' in a green header. Below this, a breadcrumb trail reads 'Home > CUDA ZONE > Tools & Ecosystem'. The 'CUDA Toolkit' section features a paragraph describing the toolkit's capabilities for C and C++ developers. Two download buttons are shown: 'DOWNLOAD CUDA 5.5 Production Release' and 'COMING SOON CUDA 6.0 Sign Up'. The right sidebar contains 'NVIDIA DEVELOPER PROGRAMS' with a 'LEARN MORE AND REGISTER' button, and 'QUICKLINKS' listing various resources like CUDA Downloads, CUDA GPUs, and NVIDIA Nsight Visual Studio Edition.

NVIDIA DEVELOPER ZONE Log In

DEVELOPER CENTERS ▸ TECHNOLOGIES ▸ TOOLS ▸ RESOURCES ▸ COMMUNITY ▸ Search DevZone

CUDA ZONE

Home > CUDA ZONE > Tools & Ecosystem

CUDA Toolkit

The NVIDIA® CUDA® Toolkit provides a comprehensive development environment for C and C++ developers building GPU-accelerated applications. The CUDA Toolkit includes a compiler for NVIDIA GPUs, math libraries, and tools for debugging and optimizing the performance of your applications. You'll also find programming guides, user manuals, API reference, and other documentation to help you get started quickly accelerating your application with GPUs.

DOWNLOAD
CUDA 5.5
Production Release

CUDA 5.5 Production Release Now Available On The [Download Page](#).

Optimized For MPI Applications

COMING SOON
CUDA 6.0
Sign Up

Dramatically Simplify Parallel Programming With CUDA 6.0 .

Unified Memory

NVIDIA DEVELOPER PROGRAMS

Get exclusive access to the latest software, report bugs and receive notifications for special events.

[LEARN MORE AND REGISTER](#)

QUICKLINKS

- CUDA Downloads
- CUDA GPUs
- NVIDIA Nsight Visual Studio Edition
- Get Started - Parallel Computing
- CUDA Tools & Ecosystem
- CUDA FAQ

0.5. Phần mềm & Lập trình

- CUDA:
 - ▣ Ứng dụng được phát triển bởi CUDA



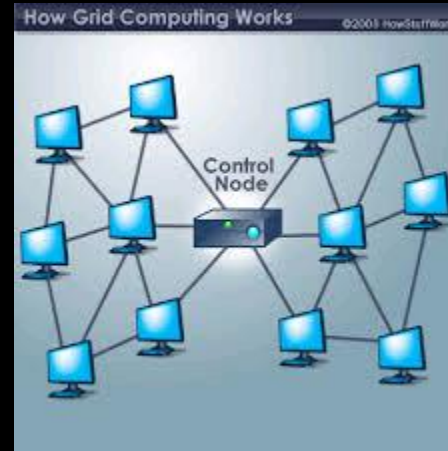


1. TỔNG QUAN VỀ TÍNH TOÁN SONG SONG



1.1.Đặt vấn đề

- Hiện nay:
 - Máy tính, thiết bị:
 - Hệ thống siêu máy tính
 - Một máy tính hỗ trợ nhiều chức năng
 - Môi trường phát triển:
 - Công cụ lập trình,
 - Ngôn ngữ lập trình
 - Thuật toán



1.1.Đặt vấn đề

- Nhu cầu:
 - Nhu cầu tính toán ngày càng tăng, đòi hỏi hiệu suất tốt hơn, cao hơn...
 - Ví dụ:
 - Mô hình hóa quá trình phát triển tự nhiên, các bài toán mô phỏng:
 - Khoa học
 - Y tế
 - Kỹ thuật
 - ...



1.1.Đặt vấn đề

- Nhu cầu:
 - Bài toán xử lý dữ liệu lớn:
 - Data Mining (Khai phá dữ liệu)
 - Computer Vision (Thị giác máy tính)
 - Image Processing (Xử lý ảnh)
 - ...
 - Bài toán dự báo: Đưa ra kết quả sau một thời gian dài phân tích và tính toán...
 - Bài toán dự báo thời tiết, bầu khí quyển được mô hình hóa thành các khối không gian 3 chiều

1.1.Đặt vấn đề

- Nhu cầu
 - Dự báo thời tiết:
 - Chia bầu khí quyển theo không gian 3 chiều mỗi khối kích thước 1mile x 1mile x 1mile
 - Ước tính khoảng 5×10^8 khối
 - Trên mỗi khối cần thực hiện khoảng 200 phép toán -> cần thực hiện khoảng 10^{11} phép toán mỗi thời điểm
 - Nếu dự báo kết quả sau 10 ngày và 10 phút update 1 lần : $10 \times 24 \times 60 / 10 = 1440$ lần -> số phép toán là 1440×10^{11}

1.1.Đặt vấn đề

- Nhu cầu:
 - Dự báo thời tiết:
 - Do đó cần 1.44×10^{14} phép tính
 - Máy tính 100MFlops => cần khoảng 1.44×10^6 seconds để thực hiện xong
 - Do đó sau 17 ngày mới biết kết quả của ngày thứ 10????



1.1.Đặt vấn đề

- Nhu cầu:
 - Bài toán dự báo thời tiết:
 - Yêu cầu tính toán trong 1 giờ = 3600s => phải có máy tính thực hiện $1.44 \times 10^{14} / 3600 = 0.4 \times 10^{11}$ phép tính / giây (=40GFlops)
 - Trong khi IBM Deep Blue chỉ có khả năng 11.38GFlop/s
 - Tháng 11/2019: Summit – Computer: 200 petaFlops/s (10^{15} FLOPS)



1.1.Đặt vấn đề

- Nhu cầu:
 - Bài toán render hình ảnh 3 chiều
 - Yêu cầu $1024 \times 1024 \times 1024 = 10^9$ pixel cần xử lý, mỗi pixel cần 200 phép tính
 - Để hình chuyển động cần 30 khung hình / s



1.1.Đặt vấn đề

- Nhu cầu:
 - Bài toán render hình ảnh 3 chiều
 - Như vậy cần có $200 \times 30 \times 10^9 = 6 \times 10^{12}$ phép tính trên một giây
 - Máy tính có khả năng xử lý 100MFlops (10^8 phép tính trên giây) máy tính này phải có 10^5 bộ vi xử lý

1.1.Đặt vấn đề

- Giải quyết vấn đề:
 - Giải quyết có 2 hướng:
 - Tăng tốc độ tính toán, hiệu suất hoạt động của máy tính => phải có thời gian và công nghệ
 - Sử dụng cùng 1 lúc nhiều bộ vi xử lý (multi-processor)
 - Bài toán được phân chia thành nhiều phần để tính toán
 - Việc giải quyết bài toán theo hướng trên được gọi là thực hiện vấn đề theo tính toán song song (parallel computing)

1.2. Tổng quan tính toán song song

- Tính toán song song là sự phát triển của tính toán tuần tự, tính toán song song tham gia cạnh tranh những cái đang tồn tại trong thế giới tự nhiên:
 - Mô phỏng khí hậu (thời tiết)
 - Phản ứng hạt nhân, hóa học
 - Hoạt động địa chất, địa chấn
 - Việc hình thành dải Ngân Hà
 - Vận động của hành tinh
 - Giao thông giờ cao điểm
 - Tuyến đường ô tô trong thành phố
 -

1.2. Tổng quan tính toán song song

- Tuy nhiên cũng có một số vấn đề không thực hiện song song được:
 - Không thể xây tường nếu chưa đổ móng
 - Không thể vào cổ áo nếu chưa hoàn thành thân áo
 - Không thể tính số hạng thứ n trong dãy Fibonacci nếu chưa tính số hạng $n-1$, $n-2$...
 - Không thể tính được $n!$ nếu chưa biết cách tính $(n-1)!$

1.2. Tổng quan tính toán song song

- Tại sao cần đến tính toán song song:
 - Tiết kiệm thời gian
 - Giải quyết được các bài toán lớn
 - Tận dụng được tài nguyên phi cục bộ (non-local) – sử dụng được các tài nguyên tính toán trên mạng diện rộng.
 - Tiết kiệm chi phí – sử dụng nhiều tài nguyên tính toán giá rẻ thay thế cho việc sử dụng một siêu máy tính.
 - Vượt qua giới hạn về lượng bộ nhớ mà máy tính sử dụng hiện nay – nhiều máy tính kết hợp -> tăng bộ nhớ không giới hạn

1.2. Tổng quan tính toán song song

■ Network:

- Sự phát triển của mạng máy tính làm thay đổi cách nhìn trong lĩnh vực tính toán
- Từ tốc độ 10Mbps, băng thông có thể vượt lên tốc độ Giga..
- Hướng mới trong lĩnh vực tính toán song song là Network Computing hay Distributed Computing
- Grid Computing, Cloud Computing, Sky Computing (Grid of Clouds)



1.2. Tổng quan tính toán song song

- Network & SuperComputer:
 - Siêu máy tính cũng biến đổi dần để tích hợp tính toán phân tán
 - Từ một máy tính gồm nhiều bộ vi xử lý biến đổi thành mạng gồm nhiều máy tính có nhiều bộ vi xử lý



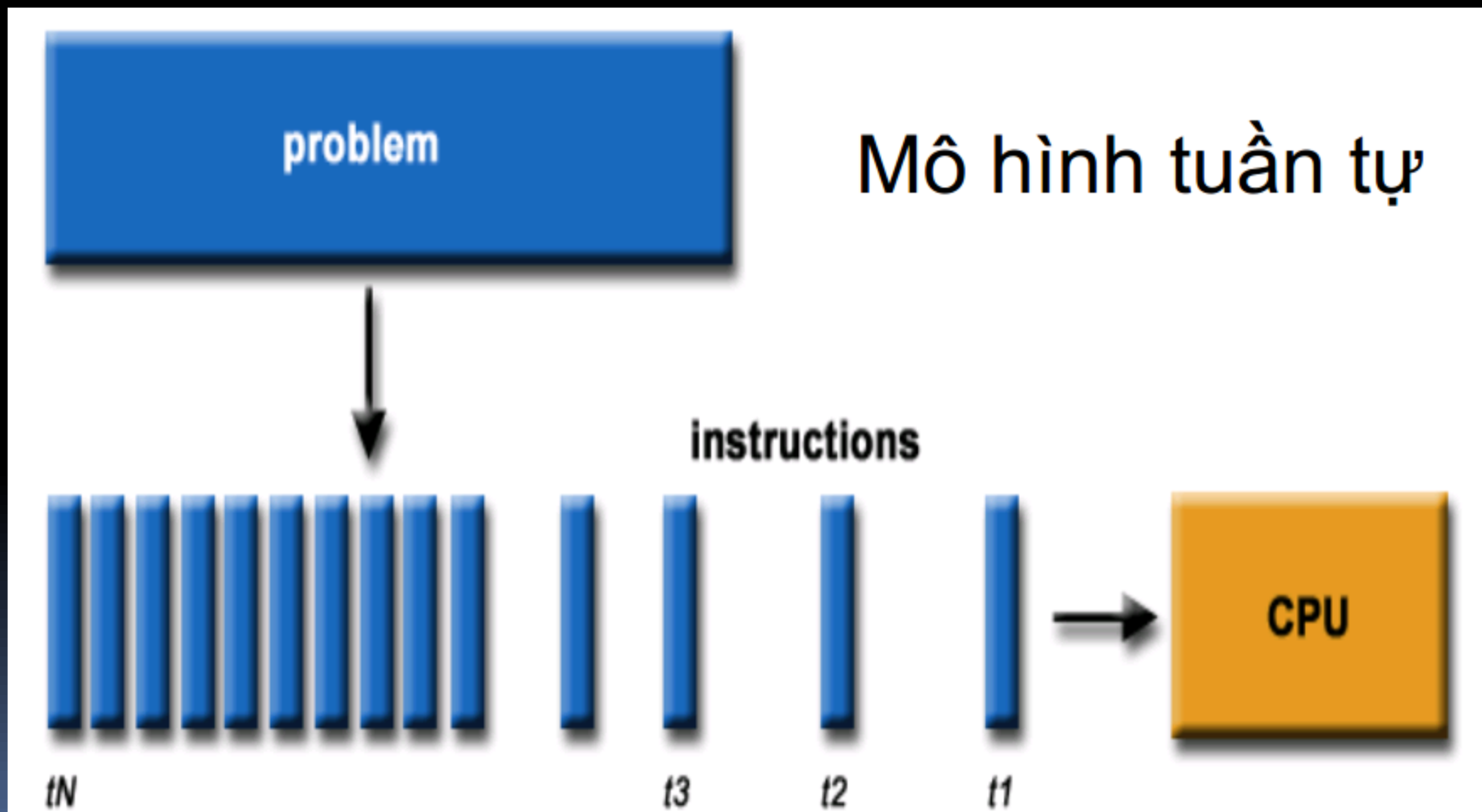
1.2. Tổng quan tính toán song song

- Do nhu cầu thực tế:
 - Sự phát triển của khoa học kỹ thuật
 - Sự phát triển của công nghệ
 - Giới hạn về kích thước, nhiệt độ,... -> Chính vì thế không thể tự do tăng mãi số lượng transistors lên được

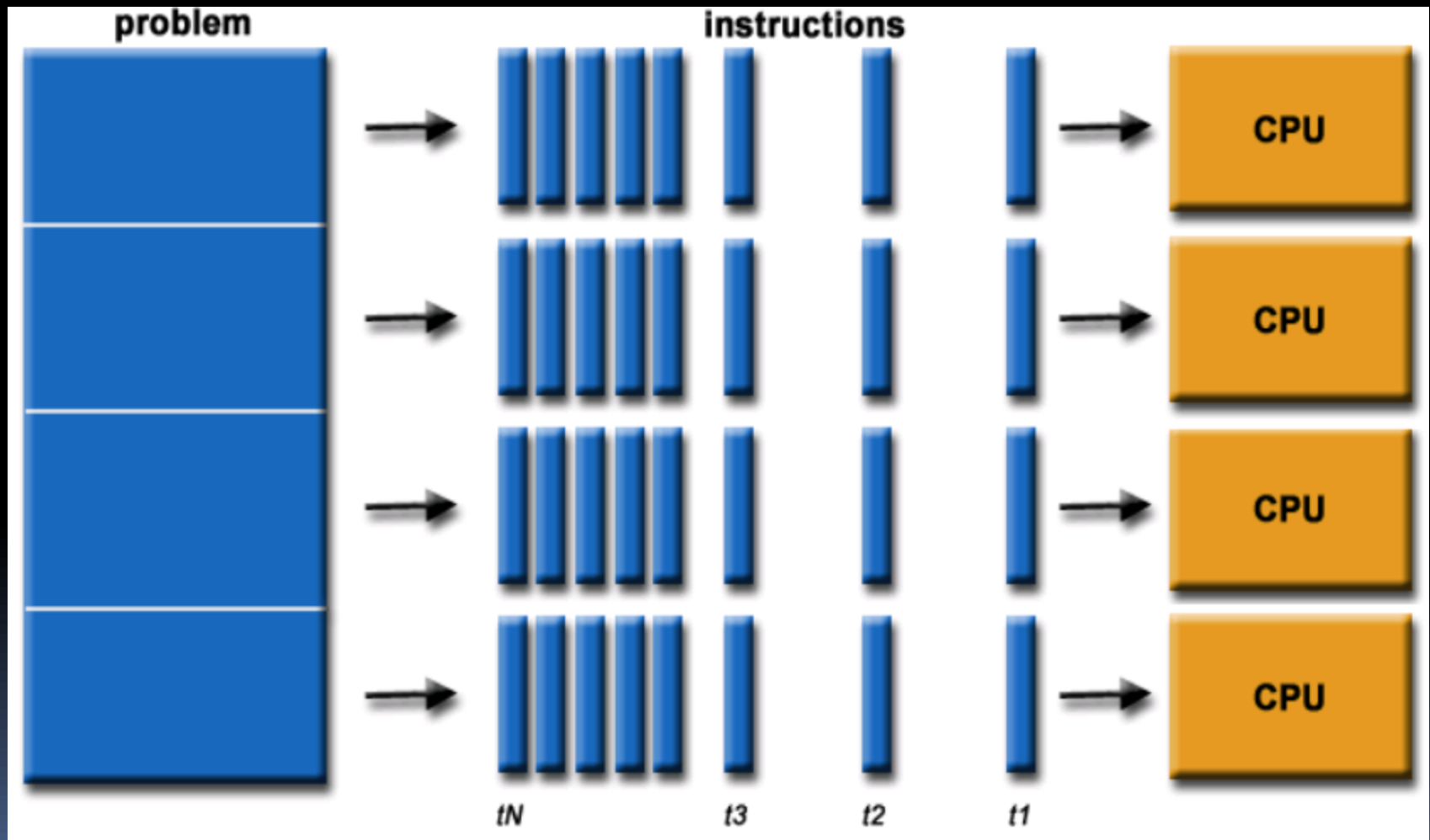
1.3. Thực hiện song song

- Nguyên tắc:
 - Phân chia công việc chính thành các công việc con, có thể thực hiện song song với nhau (khác biệt lớn nhất so với thực hiện tuần tự)
- Xây dựng hệ thống song song từ nhiều bộ xử lý riêng biệt, thực hiện các công việc song song trên các bộ xử lý đó.
- Vấn đề:
 - Phương pháp phân chia công việc
 - Môi trường thực hiện song song

1.3. Thực hiện song song



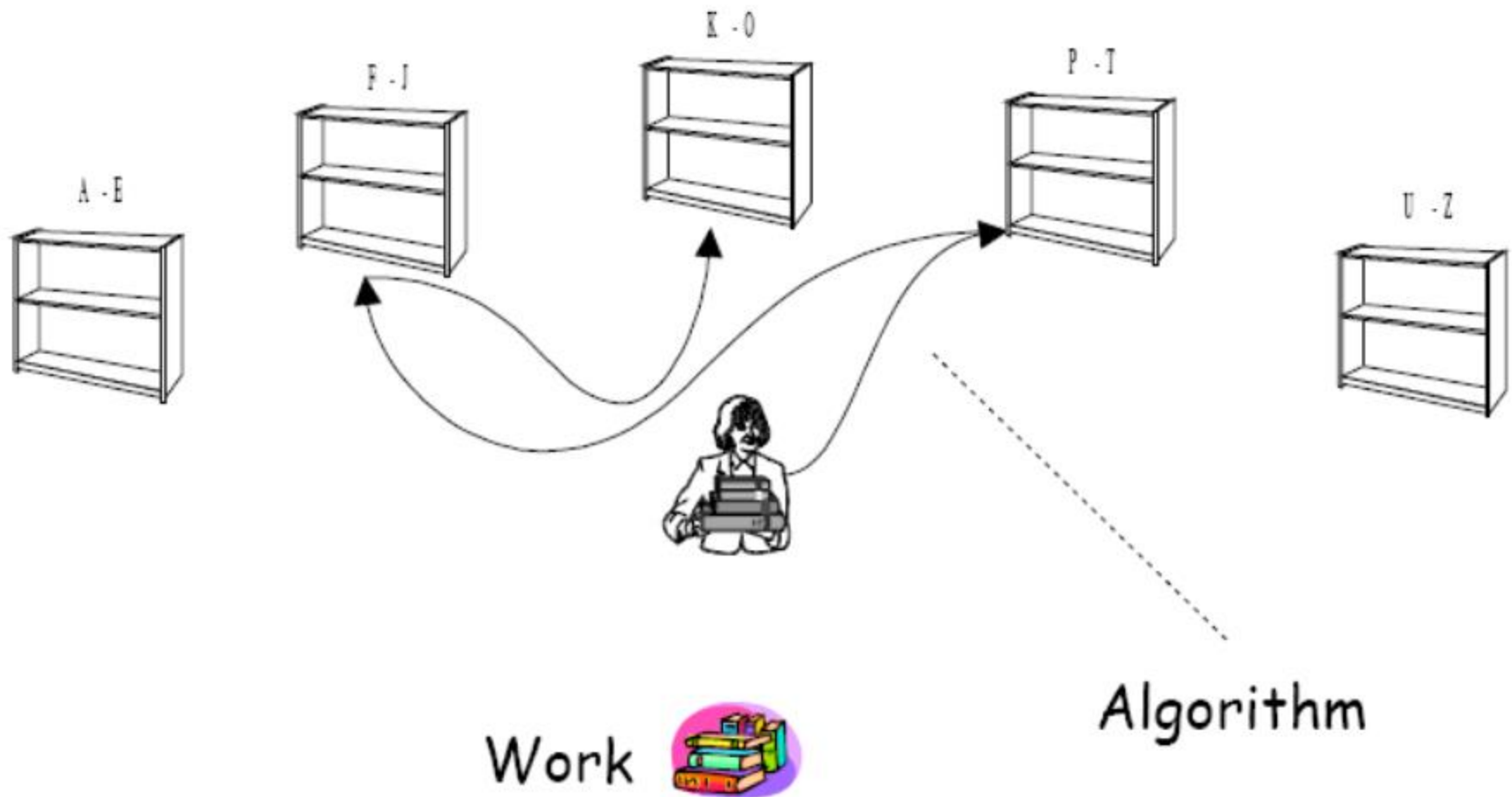
1.3. Thực hiện song song



1.4. Ví dụ trong thực tế

- Xếp sách trong thư viện:
 - Sách trong thư viện được phân loại theo chữ cái đầu tiên và sắp xếp theo thứ tự
 - Sách cùng nhóm được xếp trên cùng một giá, các giá sách đặt trong tủ sách
 - Thư viện nhập một số lượng sách lớn. Yêu cầu người thủ thư phải sắp xếp sách theo đúng nguyên tắc.
 - Cách giải quyết hiệu quả nhất?

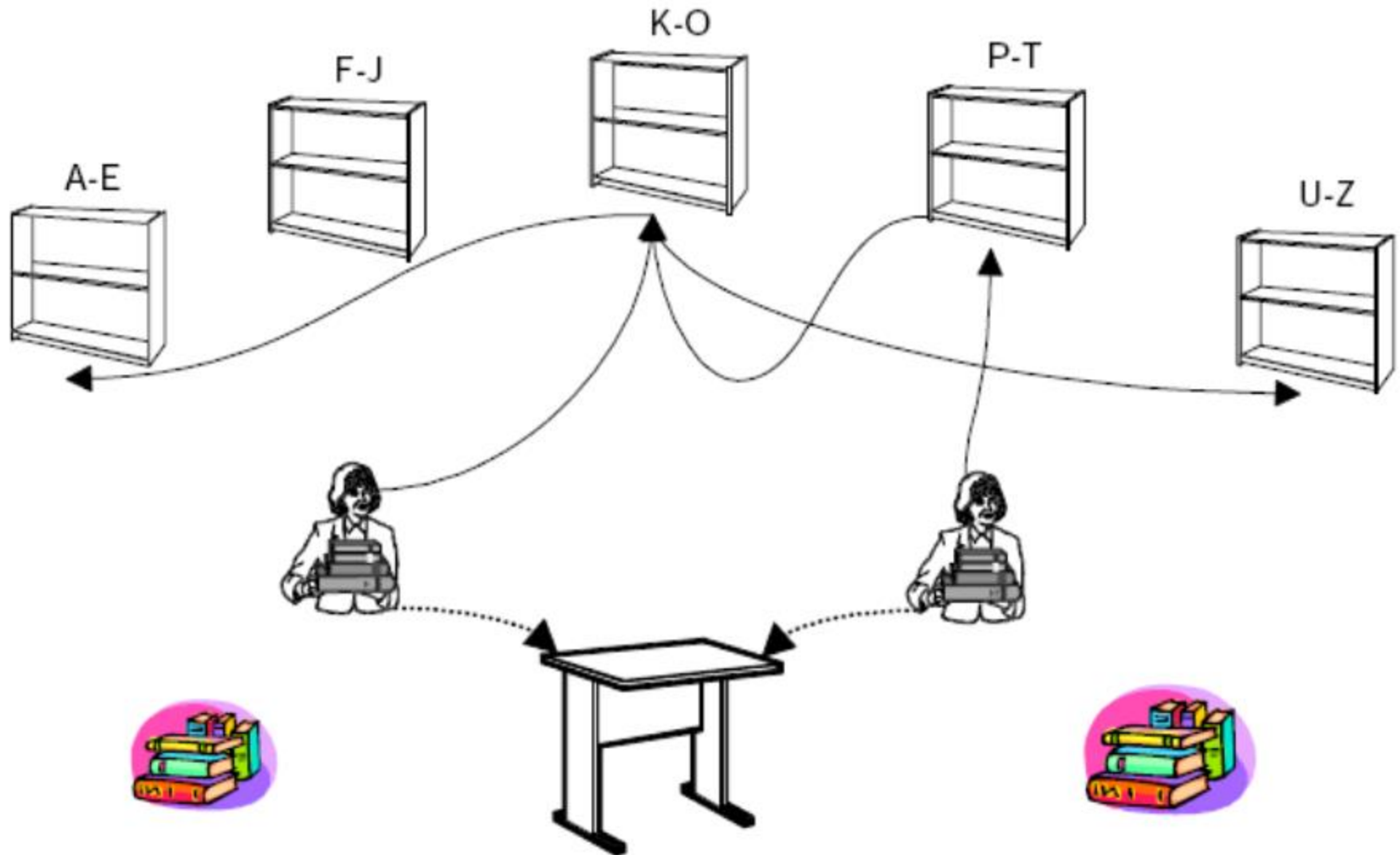
1.4. Ví dụ trong thực tế



1.4. Ví dụ trong thực tế

- Cách 1: Lấy từng cuốn sách rồi xếp vào vị trí thích hợp -> không hiệu quả.
- Cách 2: Xếp các cuốn sách theo thứ tự trước rồi mang từng chồng sách cùng vị trí đi sắp xếp -> hiệu quả hơn
- Đề xuất cách hiệu quả hơn nữa: Có nhiều thủ thư cùng thực hiện công việc đó.

1.4. Ví dụ trong thực tế



1.5. Sử dụng tính toán song song

- Được coi là tính toán cao cấp (The High end of computing), sử dụng để mô hình hóa các vấn đề khó khăn trong nhiều lĩnh vực khoa học và kỹ thuật:
 - Khí quyển, trái đất, môi trường
 - Vật lý: hạt nhân, hạt, áp lực cao, lượng tử...
 - Khoa học sinh học, công nghệ sinh học, di truyền học
 - Hóa học, khoa học phân tử
 - Địa chất, địa chấn học

1.5. Sử dụng tính toán song song

- ▣ Khoa học máy tính, toán học.
- Ứng dụng thương mại đòi hỏi công suất máy tính lớn:
 - ▣ Quản lý tập đoàn quốc gia, đa quốc gia
 - ▣ Mô hình tài chính, kinh tế
- Vấn đề xử lý dữ liệu lớn.
 - ▣ Chẩn đoán hình ảnh
 - ▣ Thiết kế dược phẩm

1.6. Ưu/nhược điểm của tính toán song song

- Ưu điểm:
 - Có thể xây dựng trên các thành phần giá rẻ
 - Rút ngắn thời gian thực hiện
 - Thực hiện công việc trong khoảng thời gian ngắn hơn nhằm tiết kiệm thời gian.
 - Thực hiện được với số lượng phép toán lớn hơn nhằm giải quyết được các bài toán lớn.
 - Hỗ trợ giải quyết đồng thời nhiều công việc
- Nhược điểm:
 - Tăng tính phức tạp của hệ thống
 - Tăng chi phí cho nhân lực vì phải tăng nhân lực (ví dụ trên)
 - Tăng chi phí mua thiết bị
 - Tăng chi phí lắp đặt hệ thống
 -

1.7.Sử dụng non-local resources

- Khái niệm:
 - Dùng nguồn tài nguyên tính toán trên mạng diện rộng, trên Internet khi nguồn tại nguyên tại chỗ cạn kiệt
 - Ví dụ:
 - SETI@home (setiathome.berkeley.edu) sử dụng 2.9 triệu máy tính kết nối Internet
 - Folding@home (folding.stanford.edu) sử dụng máy tính trên 450.000 campus

1.7. Sử dụng non-local resources

- SETI (Search The Extraerestrial Intelleigence)
 - Là một lĩnh vực khoa học phát hiện cuộc sống thông minh ngoài trái đất
 - Sử dụng kính viễn vọng radio để nghe các tín hiệu vô tuyến băng thông hẹp từ không gian.
 - Làm đài phát thanh bằng cách sử dụng 1 siêu máy tính ảo bao gồm số lượng lớn các máy tính kết nối Internet

1.7. Sử dụng non-local resources

- Folding@home:
 - Tạo môi trường tính toán phân tán nhằm tìm hiểu về sự kết xoắn và các đột biến của protein, các loại bệnh tật. Ý tưởng xây dựng tại phòng thí nghiệm của Stanford University

1.8. Giới hạn của tính toán tuần tự

- Lý do:
 - Tốc độ truyền: Phụ thuộc một cách trực tiếp vào việc dữ liệu di chuyển thông qua các phần cứng.
 - Giới hạn thu nhỏ: Công nghệ vi xử lý giúp tăng số lượng transistor đặt trên 1 con chip => còn nhiều khó khăn
 - Giới hạn kinh tế: Bộ vi xử lý mạnh giá thành cao



2. MỘT SỐ KHÁI NIỆM CƠ BẢN



2.1. Khái niệm thường gặp

- SuperComputing / High Performance Computing:
 - Lĩnh vực sử dụng máy tính nhanh và lớn để giải quyết những bài toán lớn
 - Thiết lập các thuật toán để xây dựng chương trình tính toán song song
- Node:
 - Một chiếc máy tính nằm trong chiếc hộp độc lập
 - Thông thường bao gồm nhiều CPUs/processors/cores
 - Các máy tính nối mạng cùng nhau tạo thành siêu máy tính

2.1. Khái niệm thường gặp

- CPU / Socket / Processor / Core:
 - Là các biến thể khác nhau
 - CPU – Là một thành phần thực thi duy nhất từ máy tính
 - Nhiều CPU tích hợp vào một Node
 - Một CPU riêng rẽ phân thành nhiều Core
 - CPU với nhiều Core gọi là Socket (phụ thuộc nhà sản xuất)

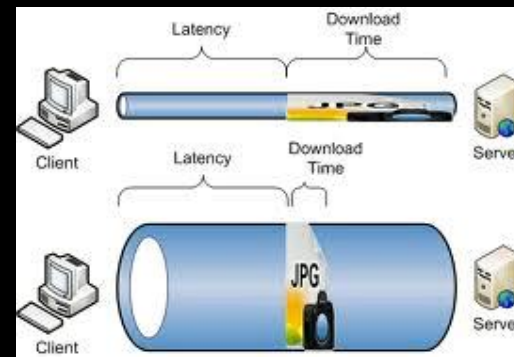
2.1. Khái niệm thường gặp

- **Massively Parallel:**
 - Song song với quy mô lớn đề cập đến yếu tố phần cứng với nhiều bộ vi xử lý.
- **Embarrassingly Parallel:**
 - Giải quyết nhiều tác vụ tương tự nhưng độc lập một cách đồng thời mà không cần đến sự phối hợp giữa chúng.
- **Scalability:**
 - Khả năng mở rộng cả phần cứng và phần mềm => chứng tỏ được speedup của hệ thống

2.1. Khái niệm thường gặp

- Task: Là một công việc cần thực hiện.
- Parallel Task: Công việc cần thực hiện song song.
- Serial Execution: Thực hiện tuần tự.
- Parallel Execution: Thực hiện song song.
- Shared Memory: Bộ nhớ dùng chung.
- Distributed Memory: Bộ nhớ phân tán.
- Communication: Sự liên lạc, trao đổi thông tin giữa các chương trình chạy song song
- Synchronization: Sự đồng bộ hóa trong thực hiện song song.

2.1. Khái niệm thường gặp



- Latency và Bandwidth là 2 tiêu chuẩn để xác định hiệu năng của một hệ thống máy tính:
 - Bandwidth: Là tốc độ mà dữ liệu có thể đưa vào từ bộ nhớ đến bộ xử lý
 - Latency: Là thời gian chờ để nhận dữ liệu khi dữ liệu đã chuyển đi
- Bandwidth cho biết số lượng phép tính thực hiện được, còn Latency là thời gian bị trễ. (Latency sẽ làm ảnh hưởng đến hệ thống).

2.1. Khái niệm thường gặp

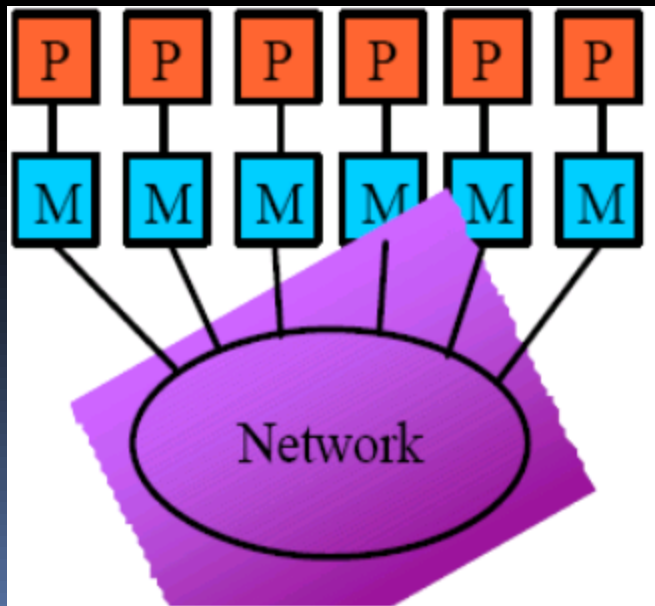
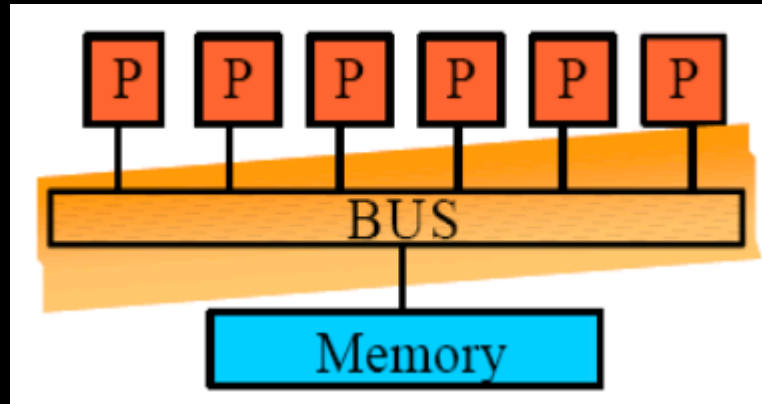
- Ví dụ Latency:

- Tính tích vô hướng của 2 vecto trên hệ thống có Latency là 100ns
- Mỗi phép toán đòi hỏi phải nạp dữ liệu vào bộ nhớ
- Do đó tốc độ tính toán bị giới hạn bởi mỗi phép tính trên là $10^{-7}s$
- Do đó máy tính giới hạn trong 1s chỉ thực hiện được $1/10^{-7}$ phép tính = 10MFlops
- Cache Memory với Latency là 1ns (bộ nhớ Cache Memory đủ chứa dữ liệu)
- Khả năng tăng lên là 1GFlops

2.1. Khái niệm thường gặp

- Máy tính Multi-core Processor ra đời với sự khai thác triệt để Cache Memory này.
- Ví dụ:
 - ▣ Dual-Core thì có 2 Cache Memory riêng biệt cho mỗi Core
 - ▣ Core 2 Dual có chung Cache Memory cho cả 2 Core

2.1. Khái niệm thường gặp



- Shared memory: Một vùng dữ liệu đơn, tất cả các processors có quyền chia sẻ bộ nhớ từ vùng dữ liệu đó. Ví dụ: SGI Origin, Sun E1000.
- Distributed memory: Mỗi processor có vùng nhớ riêng biệt, cơ chế cho phép thực hiện quá trình chuyển đổi dữ liệu giữa các processor

2.1. Khái niệm thường gặp

- Speedup (tăng tốc): $S = T_{\text{tần tự}} / T_{\text{song song}}$
- Efficiency (hiệu suất): $E = S / N_{\text{CPU}}$.
- Cost (chi phí): $C = T_{\text{kết thúc}} \times N_{\text{CPU}}$.
- Ví dụ: Có 10 CPU thực hiện song song
 - P1 thực hiện hết 10 s
 - P2 thực hiện hết 11s, ... P10 thực hiện hết 19s
 - Thời gian kết thúc công việc là 19s, chi phí thực hiện là $19 \times 10 = 190$

2.3. Các dạng song song

- Bit-level parallelism
- Instruction-level parallelism
 - VLIW
 - Pipeline
- Data parallelism
- Task parallelism

2.3. Các dạng song song

- Bit-level parallelism
 - Tính toán song song dựa trên tăng kích thước của đơn vị dữ liệu của CPU – processor word size.
 - Mô hình này phát triển từ công nghệ: VLSI – Very Large Scale Integration: tích hợp nhiều vi mạch trên chip đơn.
 - Ví dụ: cộng 2 giá trị 16 bits integer
 - Với small-scale integration CPU phải thực hiện 2 lần với từng cụm 8 bits
 - Với VLSI: CPU thực hiện cả 2 phép toán trong cùng 1 lần

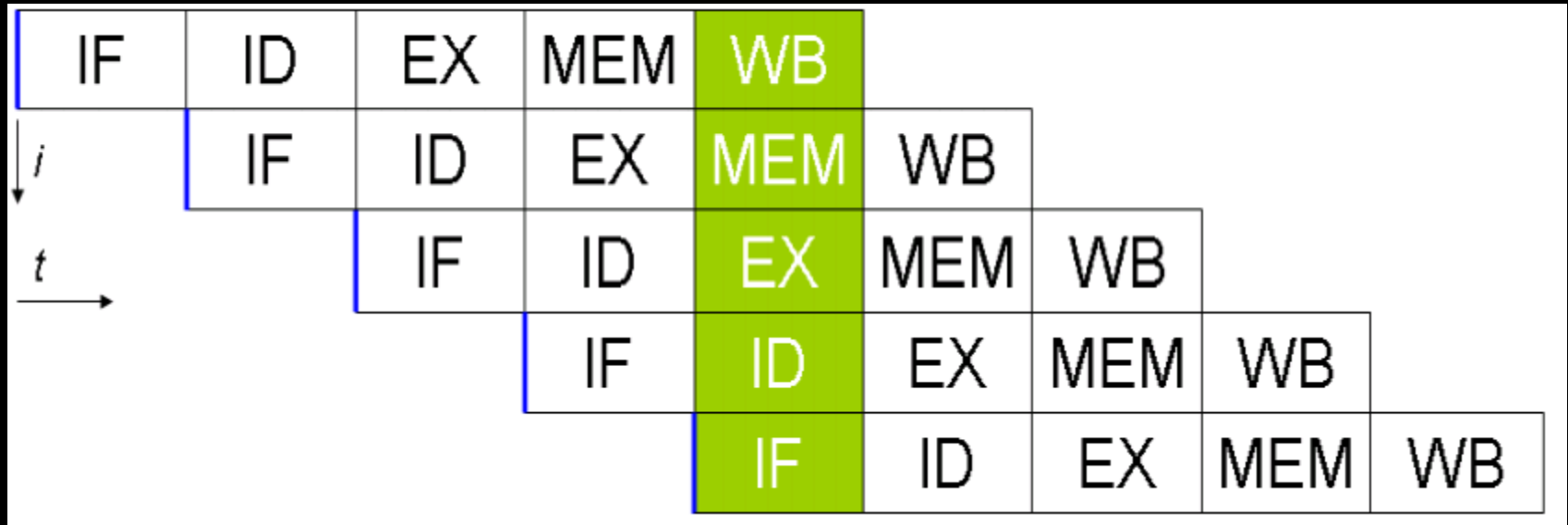
2.3. Các dạng song song

- Instruction-level parallelism:
 - Mô hình này xác định bao nhiêu phép toán có thể thực hiện đồng thời.
 - Các phép toán này thực hiện độc lập hay không phụ thuộc vào sự ràng buộc của dữ liệu thao tác.
 - Và phụ thuộc vào kiến trúc xử lý:
 - Kiến trúc độc lập: thực hiện song song – VLIW Very Long Instruction Word.
 - Kiến trúc ràng buộc: Thực hiện theo cơ chế băng chuyền Pipeline

2.3. Các dạng song song

- Instruction pipeline:
 - Các CPU hiện đại đều hỗ trợ cơ chế thực hiện Pipeline theo nhiều tầng (superscalar pipeline)
 - Dựa trên nguyên tắc chia 1 câu lệnh hoàn chỉnh thành các thành phần nhỏ hơn, mỗi phần nhỏ đó được thực hiện bởi 1 khối vi mạch riêng:
 - CISC – Complex Instruction Set Computer
 - RISC – Reduced Instruction Set Computing

2.3. Các dạng song song



- Instruction pipeline
 - IF – Instruction Fetch: Lấy mã lệnh
 - ID – Instruction Decode: Giải mã lệnh
 - EX – Execute: Thực hiện lệnh
 - MEM – Memory access: truy cập bộ nhớ
 - WB – Register Write Back: Ghi dữ liệu ra register

2.3.Data/Task parallelism

- Data parallelism: Là cơ chế phân bố dữ liệu cho các bộ xử lý sao cho các bộ xử lý thực hiện cùng một câu lệnh trên các dữ liệu song song.
- Task parallelism: Là cơ chế mà các bộ xử lý có thể thực hiện song song các câu lệnh trên các dữ liệu của mình

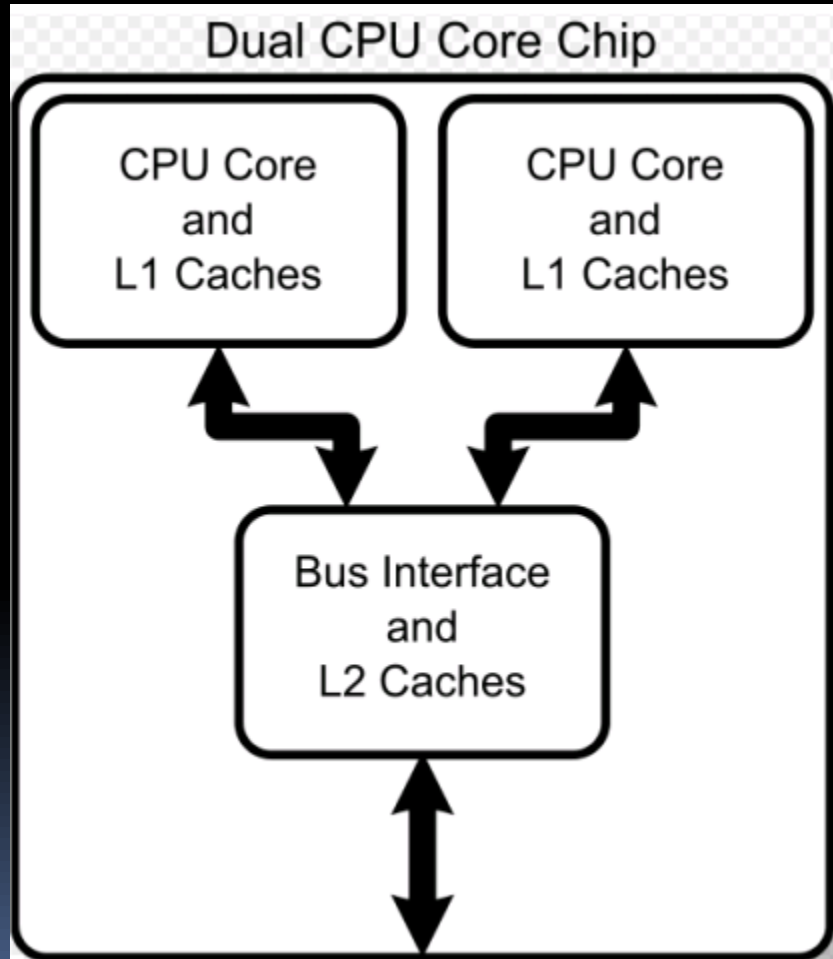


3. CÁC HỆ THỐNG TÍNH TOÁN SONG SONG

3.1. Mô hình máy tính song song

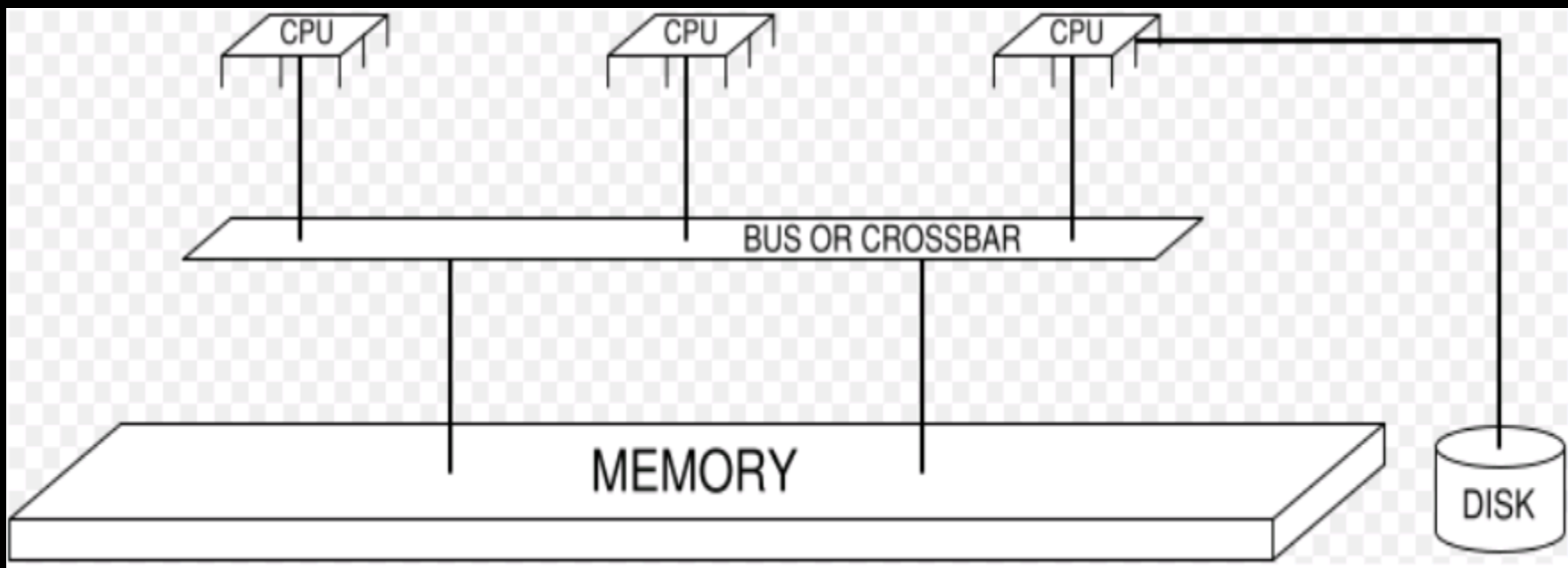
- Siêu máy tính đa nhân: Multi-core processor.
- Đa xử lý đối xứng: Symmetric multiprocessing
- Hệ phân tán: Distributed computing

3.1. Mô hình máy tính song song



- Multi-core processor:
 - Bộ xử lý trung tâm được tạo thành từ nhiều nhân độc lập (cores)
 - Các nhân thực hiện song song nhiều câu lệnh hoàn chỉnh khác nhau với các dữ liệu khác nhau (Multi-thread model)

3.1. Mô hình máy tính song song



- Symmetric multiprocessing:
 - Các nhân giống nhau kết nối cùng 1 bộ nhớ - vai trò như nhau.
 - Mỗi CPU thực hiện 1 task riêng biệt.
 - Do hạn chế về kênh truyền bus nên hệ thống này chỉ xây dựng với số nhân không lớn (32)

3.1. Mô hình máy tính song song

- Distributed computing:
 - Là hệ thống đa xử lý với bộ nhớ phân tán (Distributed memory)
 - Các bộ xử lý kết nối với nhau thông qua mạng (networking)
 - Một số dạng tiêu biểu:
 - Cluster Computing
 - Massive Parallel Processing
 - Grid Computing
 - Cloud computing



4. CÁC ỨNG DỤNG SONG SONG

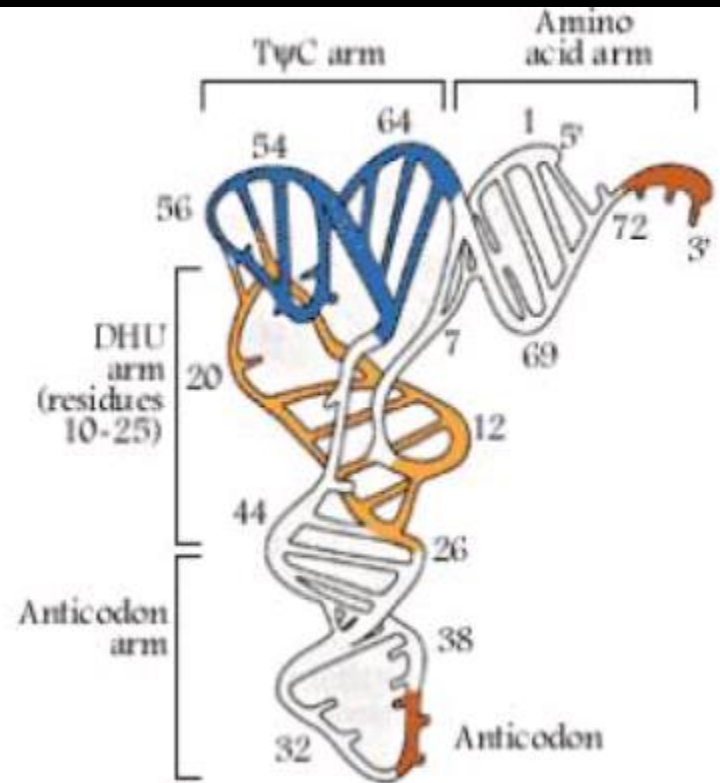
SIMULATION: MÔ PHỎNG

- Xu hướng nghiên cứu khoa học trước kia:
 - Thiết kế trên giấy dựa vào các lý thuyết
 - Thiết kế hệ thống và thực hiện các thí nghiệm
- Giới hạn:
 - Khó thực thi trong phòng thí nghiệm
 - Thực tế chi phí đắt
 - Tốc độ chậm có thể nguy hiểm
- Xu hướng hiện đại:
 - Sử dụng các máy tính hiệu năng cao để mô phỏng dựa trên các tính toán lý thuyết.

CÁC LĨNH VỰC ỨNG DỤNG

- Khoa học:
 - Mô phỏng khí hậu, vũ trụ, gen, protein....
 - Mô phỏng về đại dương, núi lửa, động đất....
- Kinh tế:
 - Mô hình kinh tế, tài chính...
 - Thực hiện giao dịch, tìm kiếm số lượng lớn...
- Quân sự:
 - Vũ khí hạt nhân...
 - Mã hóa, bảo mật

DETERMINING PHYLOGENETIC TREES



FILM INDUSTRY

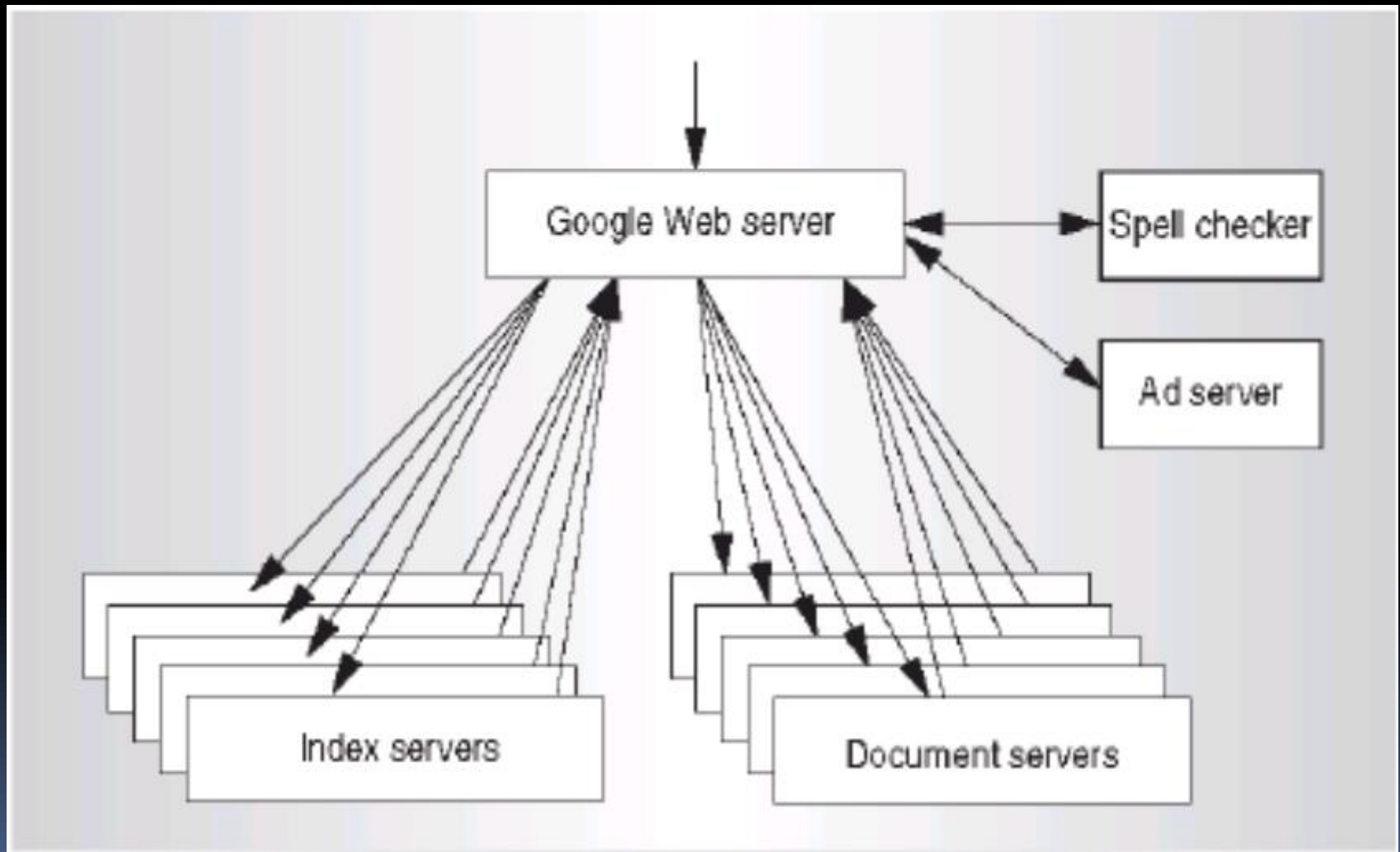


- Hiệu ứng đặc biệt cần độ phân giải cao
- Ví dụ: Lord of the Rings
 - Company: Weta Digital
 - 3200 processor cluster
 - A single scene contains:
 - 24 frames/s
 - 4996 x 3112 points with 32 (64) color/ frame encoding

SEARCH ENGINE GOOGLE

- Mỗi web search chứa vài trăm MB dữ liệu
- Khối lượng dữ liệu của Google lên đến hàng trăm terabytes.
- Mỗi web search được gọi là trivially parallel.
- Google's cluster philosophy:
 - Many cheap nodes.
 - Reliability achieved through replication in software and hardware

SEARCH ENGINE GOOGLE



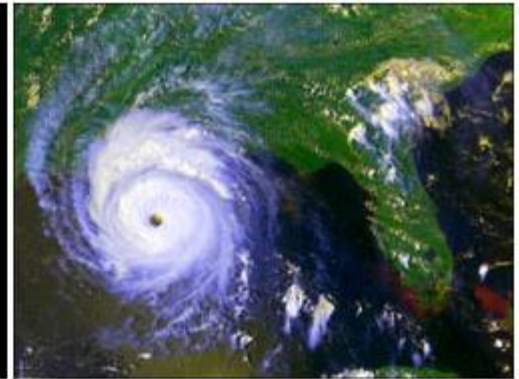
THE REAL WORLD



Galaxy Formation



Planetary Movments



Climate Change



Rush Hour Traffic

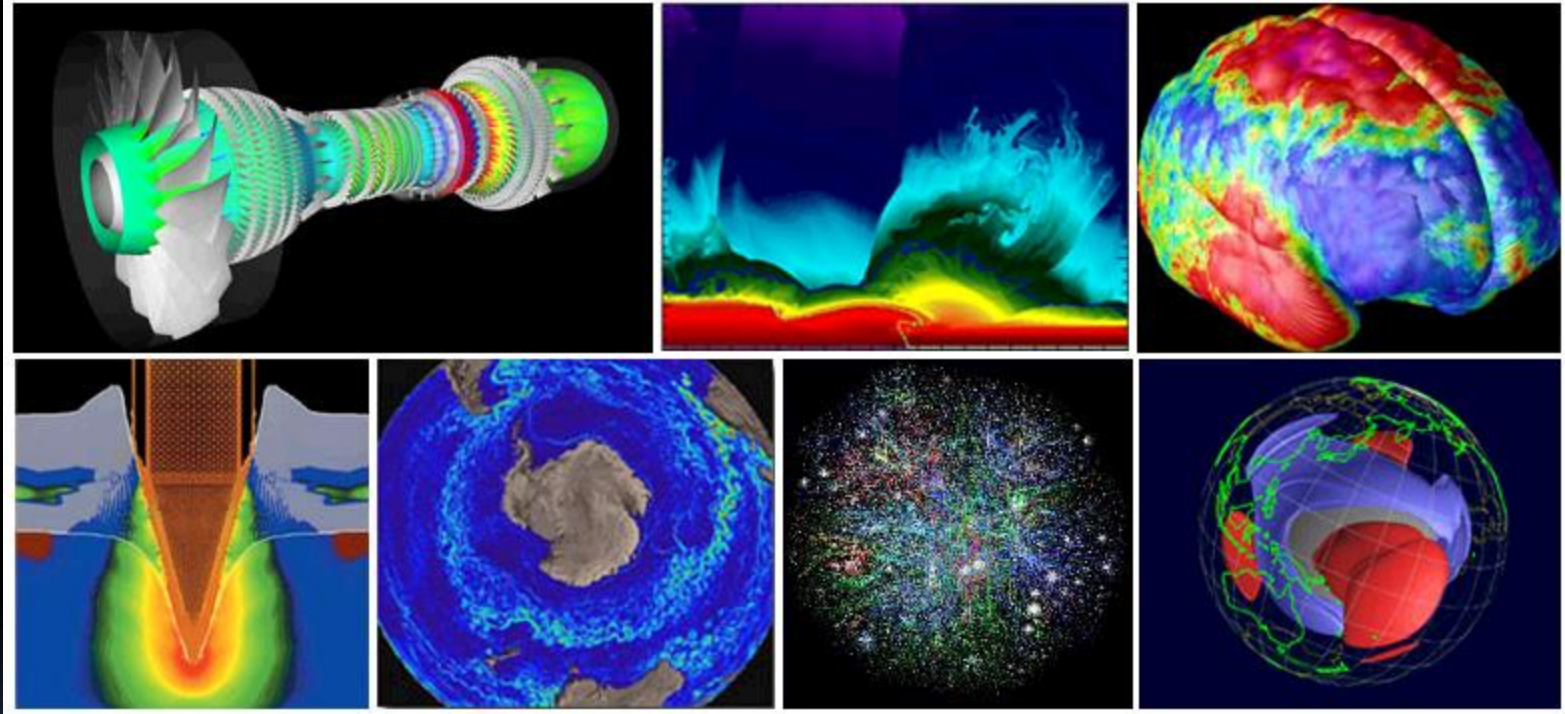


Plate Tectonics

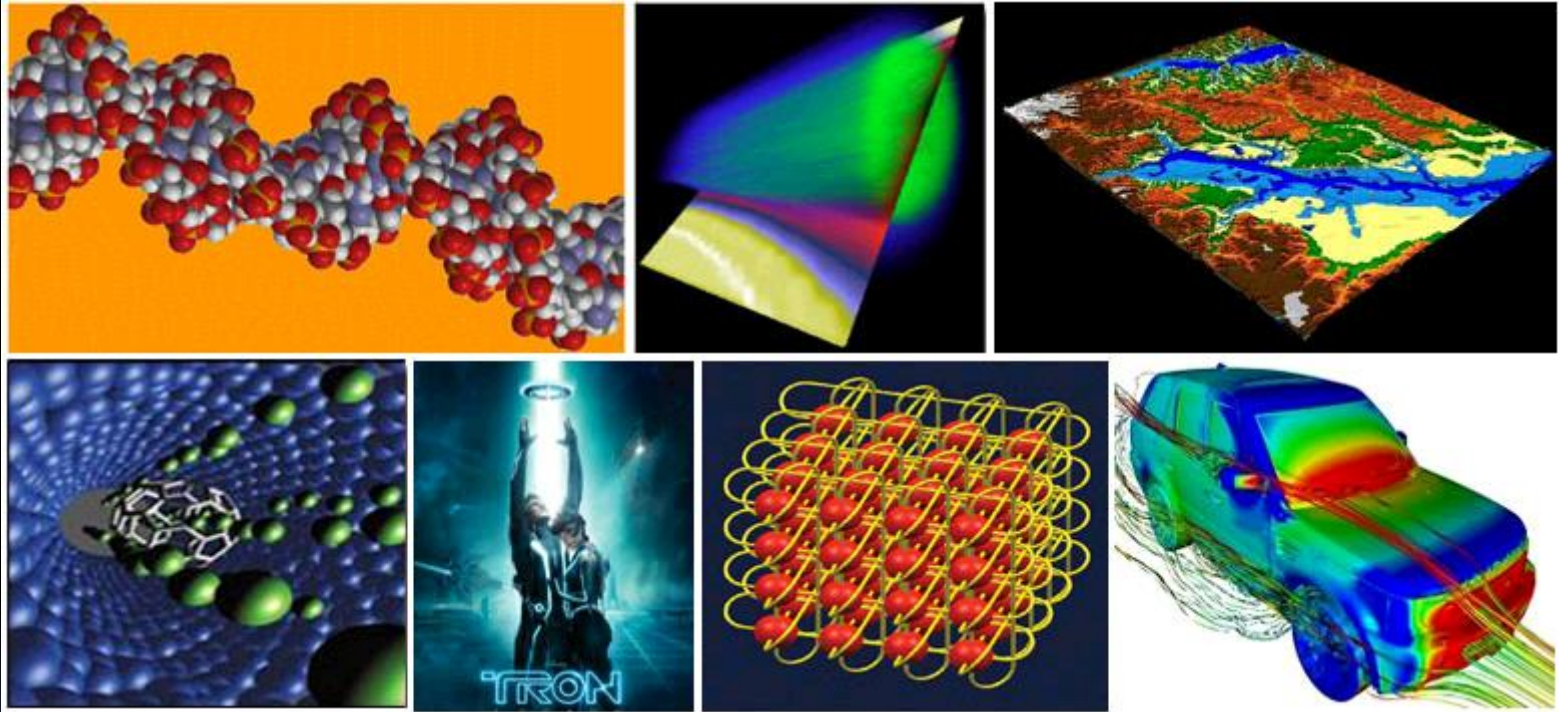


Weather

SCIENCE AND ENGINEERING



INDUSTRIAL AND COMMERCIAL



HẾT BÀI 1!!!