
# Foundation on High Performance Computing part 1 (DSSC student )

# Introduction to HPC (course P1.2) for MHPC students

Stefano Cozzini
Luca Tornatore

Trieste, 2nd October 2018



# FHPC/P1.2 course:

# Lecture 1: Introduction to HPC

Stefano Cozzini

CNR/IOM and eXact-lab srl


MHPC
Master in High Performance Computing

# Agenda for today

- Prologue: why  and where HPC ?
- What is HPC ?
  - Definitions & metrics
- What is HPC infrastructure ?
  - Supercomputers & HPC Cluster
  - CPUs and Accelerators
  - Network/storage
- HPC Concepts
  - Parallel programming paradigms
  - Evolution of paradigms
  - Ahmdal law / Gustafson law
  - Strong/weak scalability
- HOMEWORK&LABS ( for tomorrow)

SISSA ICTP

MHPC
Master in High Performance Computing

# Agenda: second part (Friday 5)

- More on HPC components

- Software stack for HPC
  - Middleware: queue systems
  - Libraries/ Compiler/ performance Tools

SISSA ICTP

MHPC
Master in High Performance Computing

# Why is HPC important ?

"The next 10 to 20 years will see computational science firmly embedded in the fabric of science – the most profound development in the scientific method in over three centuries" (US Department of Energy).

"A host of technologies are on the horizon that we cannot hope to understand, develop, or utilize without simulation" (US National Science Foundation)

SISSA ICTP

# HPC: the challenge (from EU web site)

Societal, scientific and economic needs are the drivers for the next generation of HPC - computing with exascale performance (computers capable of performing 10 to the power of 18 floating point operations per second).

- All scientific disciplines are becoming "computational" today. Modern scientific discovery requires very high computing power and capability to deal with huge volumes of data.
- Industry and SMEs are increasingly relying on the power of supercomputers to invent innovative solutions, reduce cost and decrease time to market for products and services.
- HPC is part of a global race. Many countries (USA, Japan, Russia, China, Brazil, India) have announced ambitious plans for building the next generation of HPC with exascale performance and deploying state-of-the-art supercomputers.

From https://ec.europa.eu/programmes/horizon2020/en/h2020-section/high-performance-computing-hpc

HPC capabilities are used to solve and address scientific, industrial and societal challenges.

## Health

- Development of personalised and precision medicine to provide individual and accurate patient treatment.
- Saving time and money on the development of new drugs from the initial idea phase to the final phase of reaching the market.

| development of new drugs reduced from 17 years | HPC can help save hundreds of millions |
|---|---|
| ↓ | |
| to several months | up to billions of Euros |

- Early detection of diseases and quicker diagnosis.

## Climate change & weather forecast

- Europe paid severe weather damage costs between 1970 and 2012[1].

150.000 lives — €270 billion in economic damages

- With HPC technology, climate scientists will be able to predict the size and paths of storms and floods with accuracy.
- This will allow implementation of measures such as alerting or evacuating people, thus saving human lives.

## Industry

- Reducing development time, minimising costs, optimising decision processes and producing higher quality goods and services.
- E.g. automotive industry will save time and money to develop new vehicle platforms, with improved environmental friendliness and passenger comfort and safety[2].

| development reduced from 60 months | HPC can help save |
|---|---|
| ↓ | |
| to 24 months | up to € 40 billions |

## Cybersecurity

- Enabling complex encryption technologies and better reactions to cyberattacks.
- Combined with Artificial Intelligence, HPC detects:
  - strange systems behaviour,
  - insider threats and electronic fraud,
  - very early cyberattack patterns (few hours instead of a few days)

  This will allow automated and immediate actions even before a cyberattack happens.

## Energy

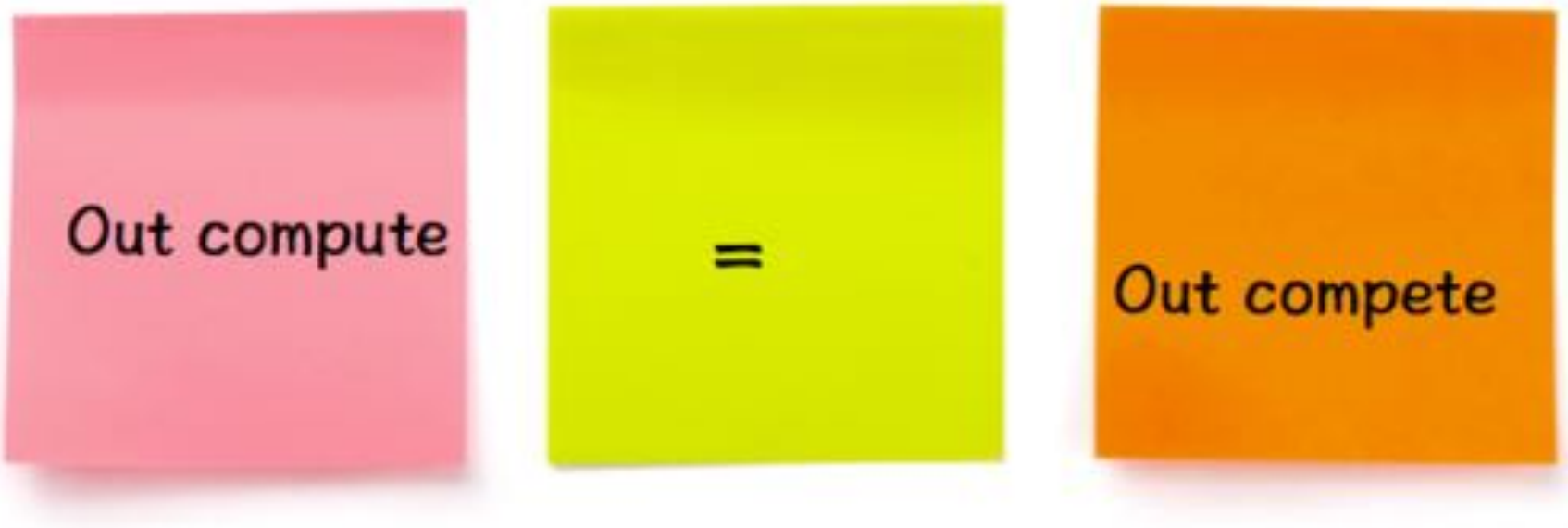
- HPC provides critical tools for example in:
  - designing renewable energy parks
  - designing high-performance photovoltaic materials,
  - optimising turbines for electricity production.
- HPC expenditure in the energy sector is projected to grow by 5% in the next years.

MHPC
Master in High Performance Computing

# Why HPC is important?


KEY ENABLING TECHNOLOGY

SISSA ICTP




***"Today, to Out-Compute is to Out-Compete".***

Image from UberCloud

MHPC
Master in High Performance Computing

# HPC stands for…

## Defining HPC (from Intersect survey)

- High Performance Computing (HPC) is the use of servers, clusters, and supercomputers – plus associated software, tools, components, storage, and services – for scientific, engineering, or analytical tasks that are particularly intensive in computation, memory usage, or data management
- HPC is used by scientists and engineers both in research and in production across industry, government and academia.

[to be continued]

SISSA ICTP

MHPC
Master in High Performance Computing

# Elements of HPC..

- use of servers, clusters, and supercomputers
  → HARDWARE
- associated software, tools, components, storage, and services
  → SOFTWARE
- scientific, engineering, or analytical tasks
  → PROBLEMS TO BE SOLVED..

ALL THE ABOVE DEFINES A COMPUTATIONAL INFRASTRUCTURE aka E-INFRASTRUCTURE

SISSA ICTP

# Elements of e-infrastructure for HPC

- E-infrastructure for HPC includes:
  - Servers/nodes/accelerators
  - High speed Networks
  - High end parallel storage
  - Middleware
  - Scientific/Technical Software
  - Research/Technical data (scientific databases, individual data…)
  - Problems to be solved

IS ALL WHAT WE NEED ?

MHPC
Master in High Performance Computing

NO

SISSA ICTP

MHPC
Master in High Performance Computing

# Last but not least: people

- Human capital is by far the most important aspect
- Two important roles:
  - HPC providers (plan/install/manage HPC resources)
  - HPC user

MIXING/INTERPLAYING ROLES INCREASES COMPETENCE LEVELS

SISSA ICTP

MHPC
Master in High Performance Computing

# From infrastructure to ecosystem

## Goal:

provide a computational ecosystem to satisfy all the different requirements posed by users

## Which kind of requirements ?

All you need to to solve their computational problems !

SISSA ICTP

MHPC
Master in High Performance Computing

# Which kind of users ?

- Computational scientists
- Industries with computational tasks

- Scientists with a lot of (different) data
- Industries with a lot of (different) data


SISSA ICTP

MHPC
Master in High Performance Computing

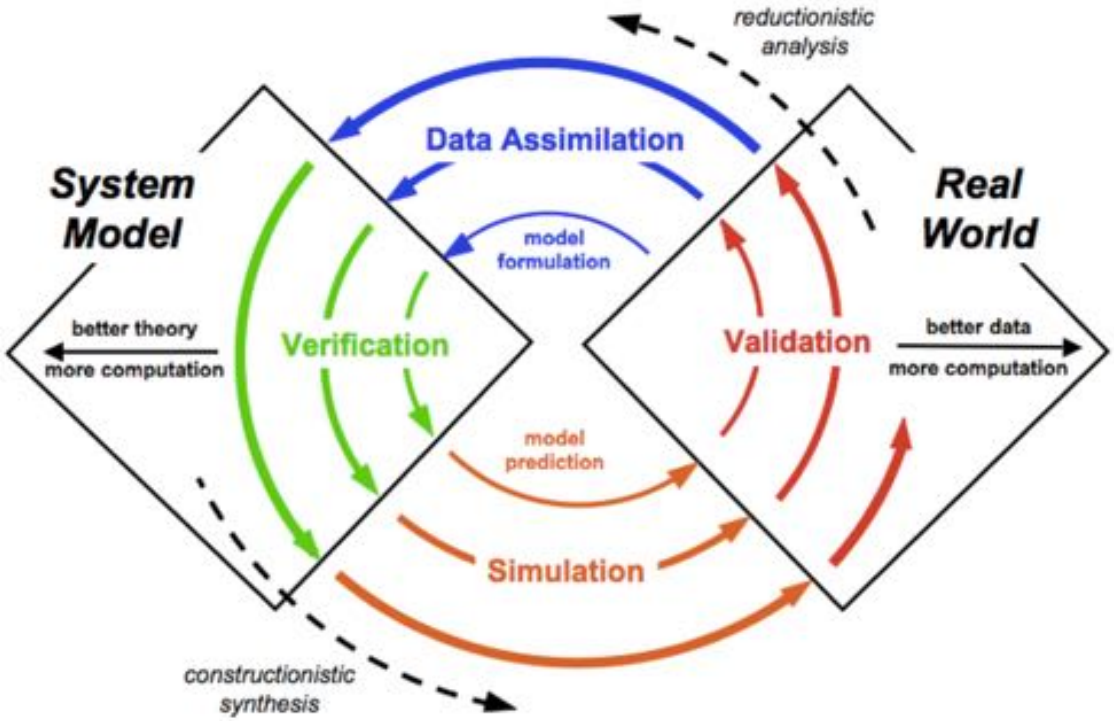
# Challenges ahead HPC (I)

- HPC skilled people

SISSA ICTP

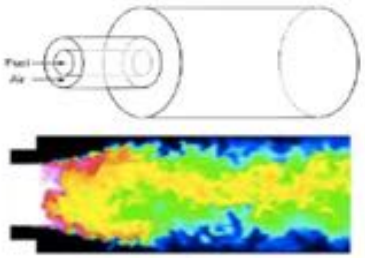
# Science research is changing

- Inference Spiral of System Science
  - As models become more complex and new data bring in more information, we require ever increasing computational

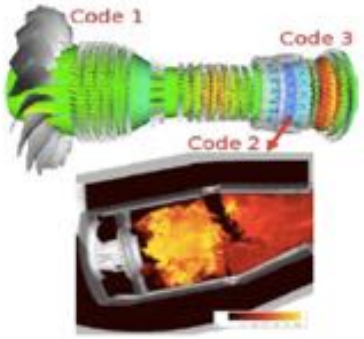


# Parallel applications require more data everyday …

- Simulation has become the way to research and develop new scientific and engineering solutions.
  - Used nowadays in leading science domains like aerospace industry, astrophysics, etc.
- Challenges related to the complexity, scalability and data production of the simulators arise.
  - Impact on the relaying IT infrastructure.

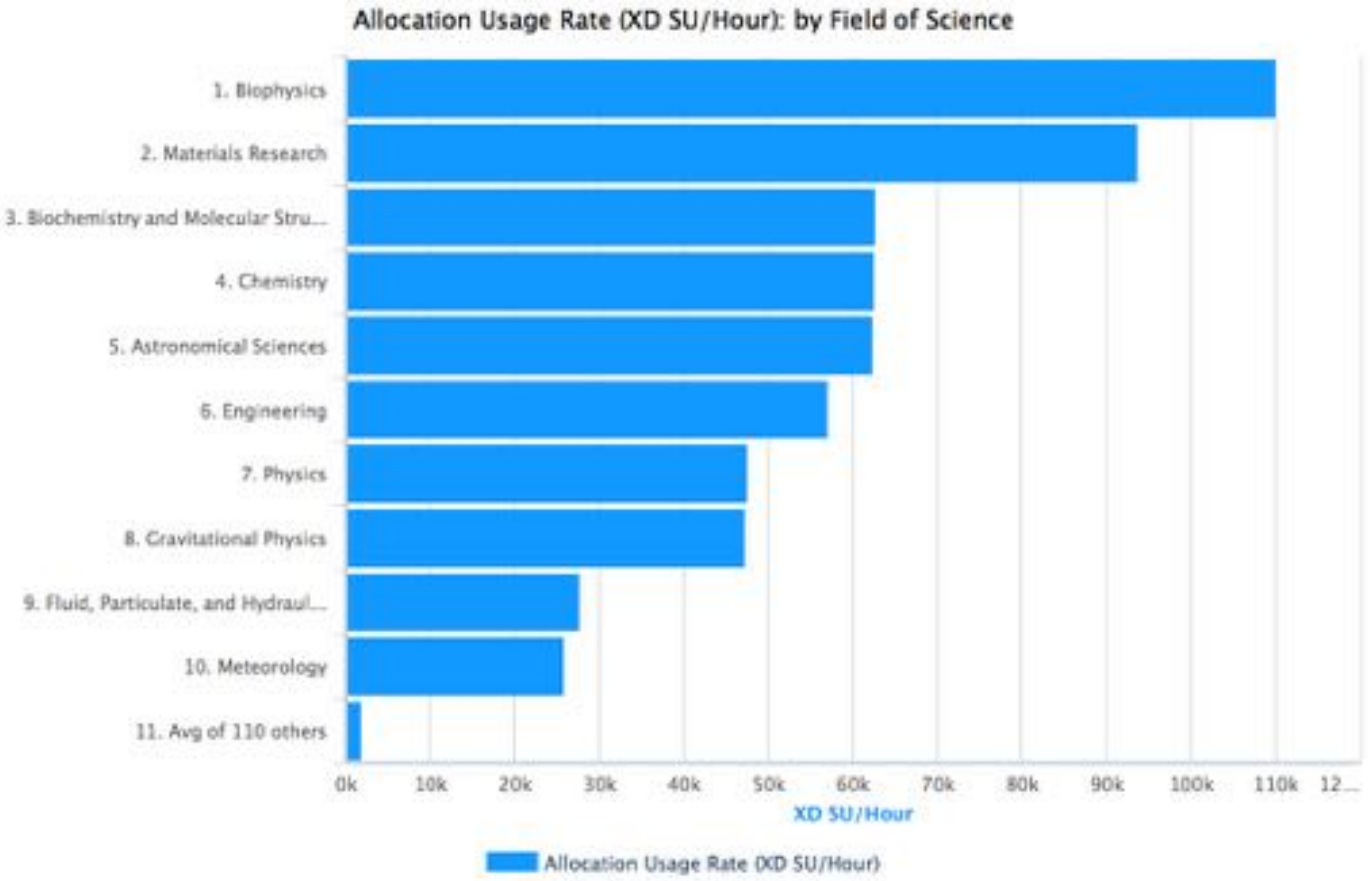


(a) State of the art in 1997




(b) State of the art in 2007

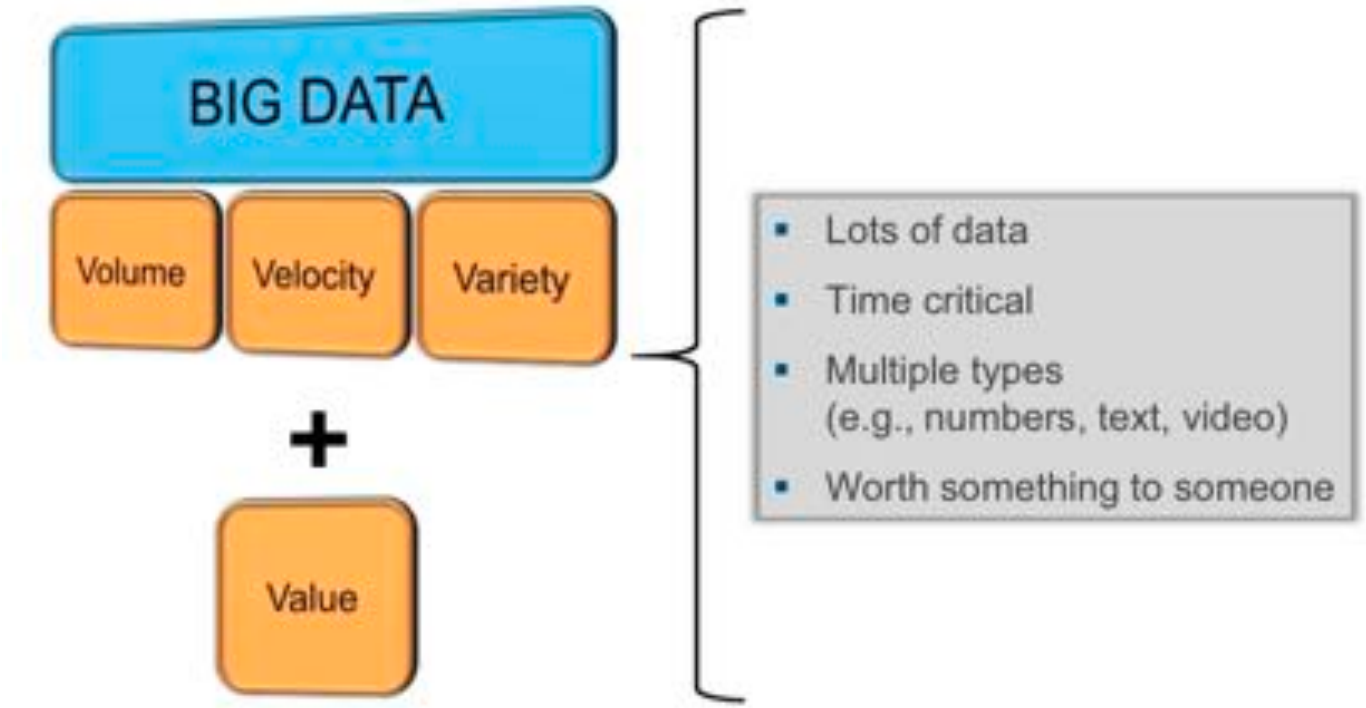
# Data generation on HPC system




Source: https://xdmod.ccr.buffalo.edu

.

# Big Data: general definition



# The 3 V's of big data..

- Velocity

  Data are produced at speed higher than the speed you are able to move/analyze and understand them..
- Variety
  - Data range from simulation to remote sensing information, from instruments to market analysis etc..
  - datasets come in a variety of data formats and span a variety of metadata standards
- Volume

  From the dawn of civilization until 2003, humankind generated five exabytes of data. Now we produce five exabytes every two days… and the pace is accelerating"

# Data-intensive science

- A "fourth paradigm" after experiment, theory, and computation..




It involves collecting, exploring, visualizing, combining, subsetting, analyzing, and using huge data collections

# Big data challenges: HPC is now HPDA

- "*BigData*" is a growing trend affecting many HPC applications touching large datacenters, and research
- Fueled by creation and availability of  many data (more complex scientific instruments and sensor networks, from "smart" power grids to the Large Hadron Collider and Square Kilometer Array)
- The proliferation of larger, The increasing transformation of certain disciplines into data-driven sciences. Biology is a notable example, but this transformation extends even to humanities disciplines such as archeology and linguistics.
- The availability of newer advanced analytics methods and tools: MapReduce/Hadoop, graph analytics, semantic analysis, knowledge discovery algorithms, and others
- The escalating need to perform advanced analytics in near-real time—a need that is causing a new wave of commercial firms to adopt HPC for the first time

MHPC
Master in High Performance Computing

# Challenges ahead HPC (I)

- HPC skilled people
- Big data !

SISSA ICTP

MHPC
Master in High Performance Computing

# Computer Performance

- It is difficult to define it properly
- “speed” / “how fast” are vague terms
- Performance as a measure again ambiguous and not clearly defined and in its interpretation
- In any case performance it’s a core to HPC as a discipline
- Let discuss it in some details

SISSA ICTP

MHPC
Master in High Performance Computing

# P stands just for PERFORMANCE ?

## Performance is not always what matters..

> to reflect a greater focus on the productivity, rather than just the performance, of large-scale computing systems, many believe that HPC should now stand for High Productivity Computing.
>
> [ from wikipedia]

- P should also stand for PROFITABILITY

SISSA ICTP

MHPC
Master in High Performance Computing

# Performance vs Productivity

- A possible definition:
  - Productivity = (application performance) / (application programming effort)
- people in HPC arena have different goals in mind thus different expectations and different definitions of productivity.

Question: Which kind of productivity are you interested in ?

SISSA ICTP

# HPC stands for... (part II)

## Defining HPC (from Intersect survey)

- Within industry, HPC can frequently be distinguished from general business computing in that companies generally will use HPC applications to gain advantage in their core endeavors – e.g., finding oil, designing automobile parts, or protecting clients' investments – as opposed to non-core endeavors such as payroll management or resource planning

MHPC
Master in High Performance Computing

# High Performance problem example:


picture from http://www.f1nutter.co.uk/tech/pitstop.php

## FUNCTIONAL PARTITIONING

Analysis of the parallel solution:

**different people are executing different tasks**

## DOMAIN DECOMPOSITION

**different people are solving the same global task but on smaller subset**


# HPC

# =

# Parallel computing

MHPC
Master in High Performance Computing

# Measuring speed of HPC systems

- How fast can I crunch numbers on my CPUs ?
- How fast can I move data around ?
  - from CPUs to memory
  - from CPUs to disk
  - from CPUs on different machines
- How much data can I store ?

SISSA ICTP

# Number crunching on CPU: what do we count ?

- Rate of [million/billions of] floating point operations per second ([M|G]flops) FLOPs/S
- Theoretical peak performance:
  - determined by counting the number of floating-point additions and multiplications that can be completed during a period of time, usually the cycle time of the machine

FLOPS=Clock-rate*Number_of_FP_ operation*Number_of_cores

# Sustained (peak) performance

Real (sustained) performance: a measure

measured by taking the time the code requires to run

$$\frac{\text{(Number\_of\_floating\_point\_operations of the code )}}{\text{Time measured}}$$

-Number_of_floating_point_operations not easy to be defined for real application:

-benchmarks are available for that..

-Top500 list uses HPL Linpack:

Sustained peak performance is what's matter in TOP500

# TOP 500 List

- The TOP500 list www.top500.org
  - published twice a year from 1993
    - ISC conference in Europe (June)
    - Supercomputing conference in USA (November)
  - List the most powerful computers in the world
  - yardstick: Linpack benchmark (LU – decomposition)

# HPL: some details

From http://icl.cs.utk.edu/hpl/index.html:

The code solves a uniformly random system of linear equations and reports time and floating-point execution rate using a standard formula for operation count.

Number_of_floating_point_operations = $2/3n^3 + 2n^2$ (n=size of the system)

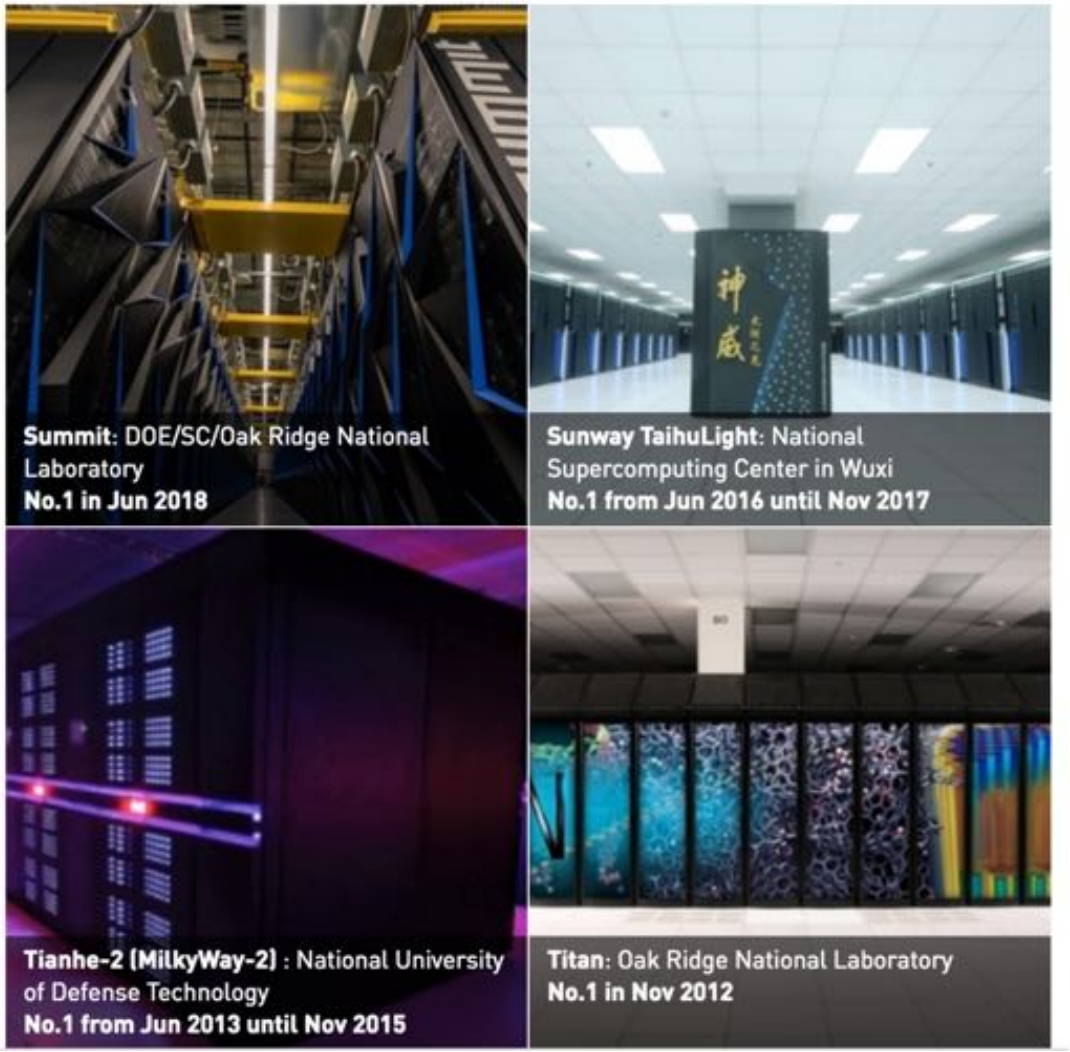
```
================================================================================
T/V                N    NB     P     Q               Time                 Gflops
--------------------------------------------------------------------------------
WR03R2L2       86000  1024     2     1             191.06              2.219e+03
--------------------------------------------------------------------------------
||Ax-b||_oo/(eps*(||A||_oo*||x||_oo+||b||_oo)*N)=        0.0043644 ...... PASSED
```

# HPL&TOP500

- For each machine the following numbers are reported using HPL:
  - Rmax: the performance in GFLOPS for the largest problem run on a machine.
  - Rpeak: the theoretical peak performance GFLOPS for the machine.
  - The measure of the power required to run the benchmark

MHPC
Master in High Performance Computing

# And the winner is…




SISSA ICTP

MHPC
Master in High Performance Computing

#1

# Summit - IBM Power System AC922, IBM POWER9 22C 3.07GHz, NVIDIA Volta GV100, Dual-rail Mellanox EDR Infiniband

| | |
|---|---|
| **Site:** | DOE/SC/Oak Ridge National Laboratory |
| **System URL:** | http://www.olcf.ornl.gov/olcf-resources/compute-systems/summit/ |
| **Manufacturer:** | IBM |
| **Cores:** | 2,282,544 |
| **Memory:** | 2,801,664 GB |
| **Processor:** | IBM POWER9 22C 3.07GHz |
| **Interconnect:** | Dual-rail Mellanox EDR Infiniband |
| **Performance** | |
| **Linpack Performance (Rmax)** | 122,300 TFlop/s |
| **Theoretical Peak (Rpeak)** | 187,659 TFlop/s |
| **Nmax** | 13,989,888 |
| **HPCG [TFlop/s]** | 2,925.75 |
| **Power Consumption** | |
| **Power:** | 8,805.50 kW (Submitted) |

MHPC
Master in High Performance Computing

# Top 1%

| 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<br>DOE/SC/Oak Ridge National Laboratory<br>United States | 2,282,544 | 122,300.0 | 187,659.3 | 8,806 |
| 2 | **Sunway TaihuLight** - Sunway MPP, Sunway SW26010 260C 1.45GHz, Sunway , NRCPC<br>National Supercomputing Center in Wuxi<br>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 , IBM<br>DOE/NNSA/LLNL<br>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 , NUDT<br>National Super Computer Center in Guangzhou<br>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 , Fujitsu<br>National Institute of Advanced Industrial Science and Technology (AIST)<br>Japan | 391,680 | 19,880.0 | 32,576.6 | 1,649 |

# First EU machine..

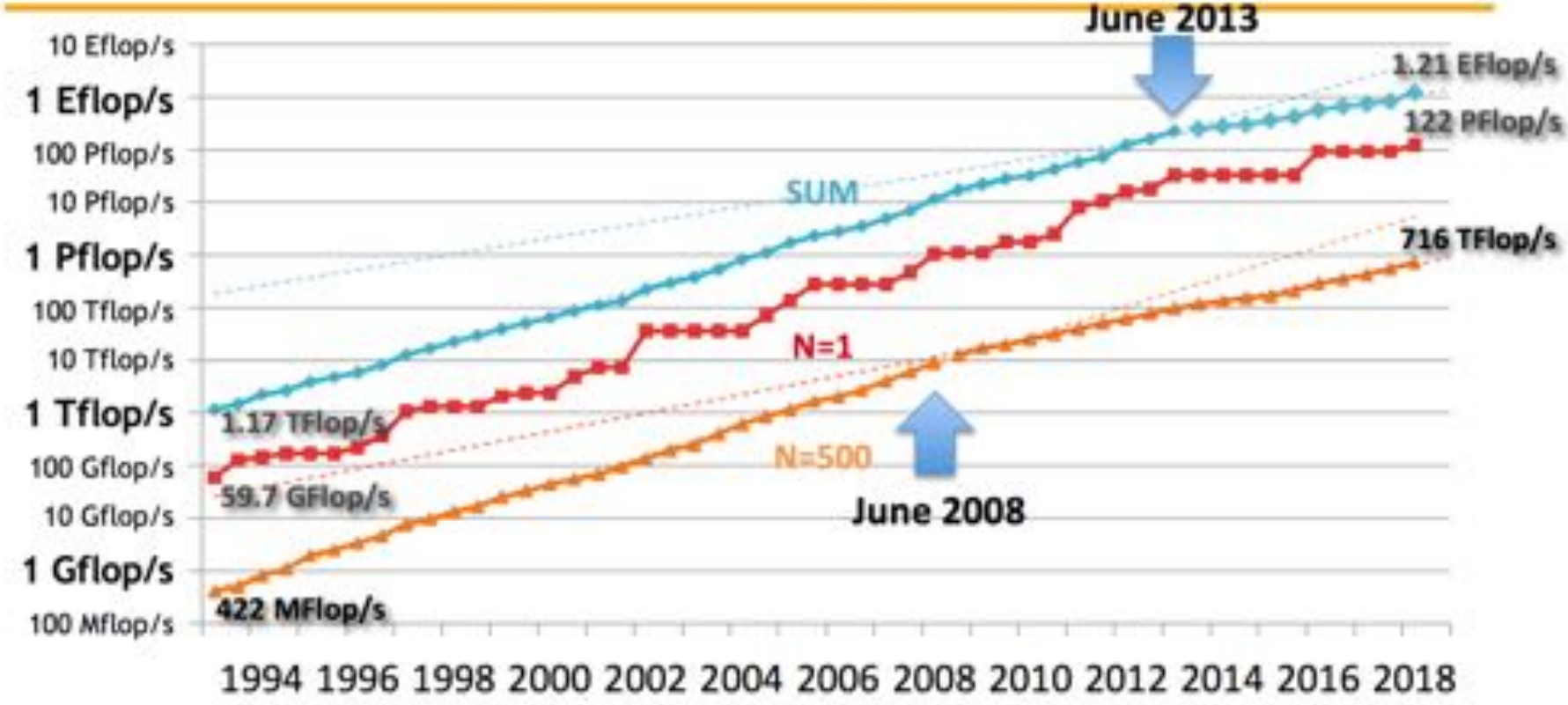
## Piz Daint - Cray XC50, Xeon E5-2690v3 12C 2.6GHz, Aries interconnect , NVIDIA Tesla P100

| | |
|---|---|
| **Site:** | Swiss National Supercomputing Centre (CSCS) |
| **System URL:** | http://www.cscs.ch/computers/piz_daint_piz_dora/index.html |
| **Manufacturer:** | Cray Inc. |
| **Cores:** | 361,760 |
| **Memory:** | 340,480 GB |
| **Processor:** | Xeon E5-2690v3 12C 2.6GHz |
| **Interconnect:** | Aries interconnect |
| **Performace** | |
| **Linpack Performance (Rmax)** | 19,590 TFlop/s |
| **Theoretical Peak (Rpeak)** | 25,326.3 TFlop/s |
| **Nmax** | 3,569,664 |
| **HPCG [TFlop/s]** | 470.0 |
| **Power Consumption** | |
| **Power:** | 2,271.99 kW (Optimized: **1631.13** kW) |
| **Power Measurement Level:** | 3 |
| **Measured Cores:** | 361,760 |
| **Software** | |
| **Operating System:** | Cray Linux Environment |

MHPC
Master in High Performance Computing

# PERFORMANCE DEVELOPMENT

TOP 500




SISSA

# Sustained peak performance on real scientific codes

Blue-waters at NCSA: 22,640 AMD 6276 processors
Theoretical peak performance: 13 Petaflops
Sustained performance on real scientific codes:..

| Scientific code | Number of cores | Performance achieved(PF) | runtime (hour) |
|---|---|---|---|
| VPIC | 22528 | 1.25 | 2.5 |
| PPM | 21417 | 1.23 | ~ 1 |
| QMCPACK | 22500 | 1.037 | ~1 |
| SPECF3MD | 21675 | >1 | Not reported |
| WRF | 8192 | 0,160 | <0.50 |

http://www.cray.com/sites/default/files/resources/XE6-NCSA-PFApplications-0514.pdf

# Why Performance degradation ?

- HPC system is unable to exploit all the resources all of the time
- Many different causes and many parts of the HPC are responsible all together
- At abstract level four important factors:
  - Starvation
  - Latency
  - Overhead
  - Waiting for Contention =>SLOW

# Starvation

- Happens when sufficient work is not available at any instance in time to support issuing instructions to all functional units every cycle.
- Typical case:
  - Not enough parallel work for all processors/components
  - Parallel work not evenly distributed among all processors/components (load is not balanced)

MHPC
Master in High Performance Computing

# Latency

- Time it takes for information to move from one part of the system to the other.
- Typical cases:
  - Memory access
  - Data transfer between separate nodes
- Lot of tricks to hide latency (see next lectures)

SISSA ICTP

MHPC
Master in High Performance Computing

# Overhead

- The amount of additional work needed beyond that which is actually required to perform the computation.
- Typical cases:
  - Time to spawn and synchronize parallel tasks
  - Other kind of operation not directly associated to the computation
- The above operations steals resources to the computation and should be minimized

SISSA ICTP

# Waiting for contention

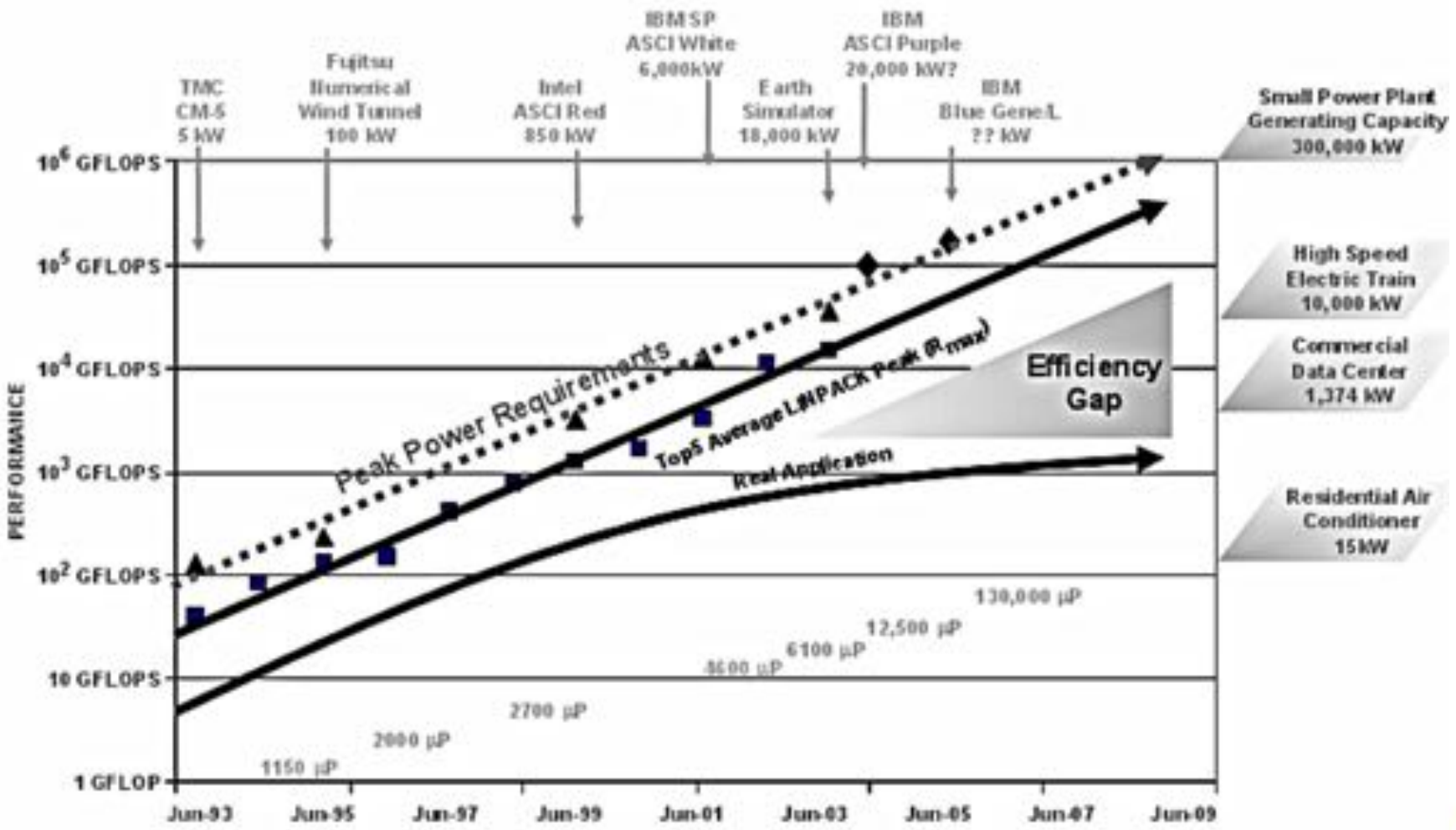
- Two or more request are made at the same time on the same resource (either HW or SW)..
- Typical cases:
  - Two task writing on the same disk and/or sending message to the same memory location at the same time
- Generally such events are not predictable and so difficult to avoid and to optimize.

MHPC
Master in High Performance Computing

# Challenges ahead HPC (III)

- HPC skilled people !
- Big data ! !
- Sustained performance not just HPL !!!

SISSA ICTP


TMC CM-5 5 kW
Fujitsu Numerical Wind Tunnel 100 kW
Intel ASCI Red 850 kW
IBM SP ASCI White 6,000kW
Earth Simulator 18,000 kW
IBM ASCI Purple 20,000 kW?
IBM Blue Gene/L ?? kW
Small Power Plant Generating Capacity 300,000 kW
High Speed Electric Train 10,000 kW
Commercial Data Center 1,374 kW
Residential Air Conditioner 15kW
PERFORMANCE
10^6 GFLOPS
10^5 GFLOPS
10^4 GFLOPS
10^3 GFLOPS
10^2 GFLOPS
10 GFLOPS
1 GFLOP
Peak Power Requirements
Top5 Average LINPACK Peak (Rmax)
Real-Application
Efficiency Gap
130,000 µP
12,500 µP
6100 µP
3600 µP
2700 µP
2000 µP
1150 µP
Jun-93
Jun-95
Jun-97
Jun-99
Jun-01
Jun-03
Jun-05
Jun-07
Jun-09

MHPC
Master in High Performance Computing

# How much energy do we need to run an exascale machine ?

- A lot !

- Exercise: compute how much energy you need to reach 1exaflop  using #1 in 2012 /2015/2018

SISSA ICTP

MHPC
Master in High Performance Computing

# Top500&Green500

- Over the last years , energy efficiency increased substantially
- BUT Exaflops machine  today  not  yet sustainable in term of energy !
- Sustainability is set to 20MW !
- Though to be reached in 2018 now postponed to 2023
- Green500 is the Top500 reorder according to energy efficiency…

SISSA ICTP

# The Ultimate Goal of "The Green500"

- Raise awareness of energy efficiency in supercomputing.
  - Drive energy efficiency as a first-order design constraint (on par with FLOPS).

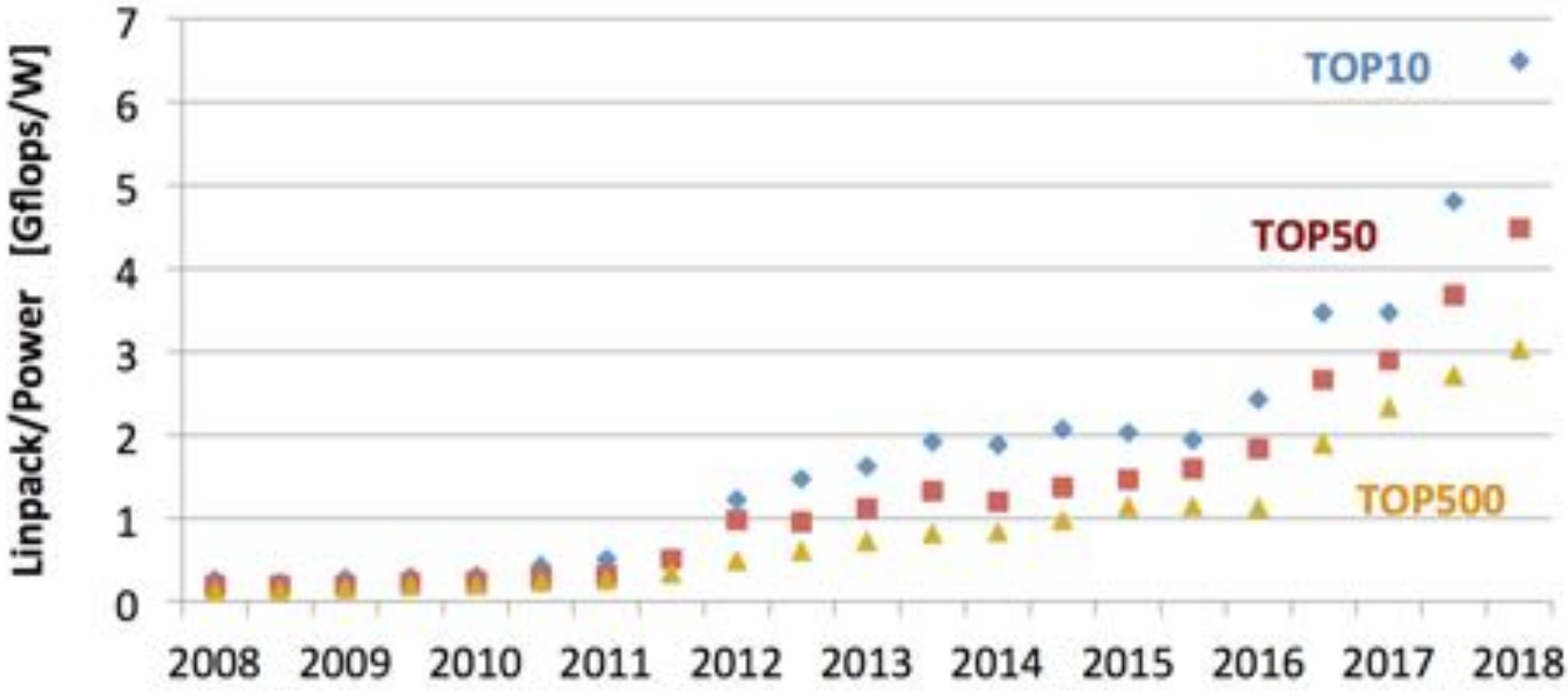
Encourage fair use of the list rankings to promote energy efficiency in high-performance computing systems.

# Top Green500 June 2018

| Rank | TOP500 Rank | System | Cores | Rmax (TFlop/s) | Power (kW) | Power Efficiency (GFlops/watts) |
|---|---|---|---|---|---|---|
| 1 | 359 | **Shoubu system B** - ZettaScaler-2.2, Xeon D-1571 16C 1.3GHz, Infiniband EDR, PEZY-SC2 , **PEZY Computing / Exascaler Inc.**<br>Advanced Center for Computing and Communication, RIKEN<br>Japan | 794,400 | 857.6 | 47 | 18.404 |
| 2 | 419 | **Suiren2** - ZettaScaler-2.2, Xeon D-1571 16C 1.3GHz, Infiniband EDR, PEZY-SC2 , **PEZY Computing / Exascaler Inc.**<br>High Energy Accelerator Research Organization /KEK<br>Japan | 762,624 | 798.0 | 47 | 16.835 |
| 3 | 385 | **Sakura** - ZettaScaler-2.2, Xeon E5-2618Lv3 8C 2.3GHz, Infiniband EDR, PEZY-SC2 , **PEZY Computing / Exascaler Inc.**<br>PEZY Computing K.K.<br>Japan | 794,400 | 824.7 | 50 | 16.657 |
| 4 | 227 | **DGX SaturnV Volta** - NVIDIA DGX-1 Volta36, Xeon E5-2698v4 20C 2.2GHz, Infiniband EDR, NVIDIA Tesla V100 , **Nvidia**<br>NVIDIA Corporation<br>United States | 22,440 | 1,070.0 | 97 | 15.113 |
| 5 | 1 | **Summit** - IBM Power System AC922, IBM POWER9 22C 3.07GHz, NVIDIA Volta GV100, Dual-rail Mellanox EDR Infiniband , **IBM**<br>DOE/SC/Oak Ridge National Laboratory<br>United States | 2,282,544 | 122,300.0 | 8,806 | 13.889 |

# POWER EFFICIENCY

TOP 500



MHPC
Master in High Performance Computing

# Challenges ahead HPC (IV)

- HPC skilled people !
- Big data ! !
- Sustained performance not just HPL !!
- Energy !!

SISSA ICTP

MHPC
Master in High Performance Computing

**Exercise 1 : compute Theoretical Peak performance for your laptop/desktop: (mandatory for MHPC&DSCC optional)**

Identify the CPU
Identify the frequency
Identify the number of floating point for cycle
Identify how many cores
Put all together in one single number and tell me

**Exercise 2 : compute sustained Peak performance for your cell-phone (mandatory for all MHPC&DSSC)**

Identify an app to run HPL
Run it
Tune it and get the best number you can…

SISSA ICTP

# Moving data around: bits and Mb/sec

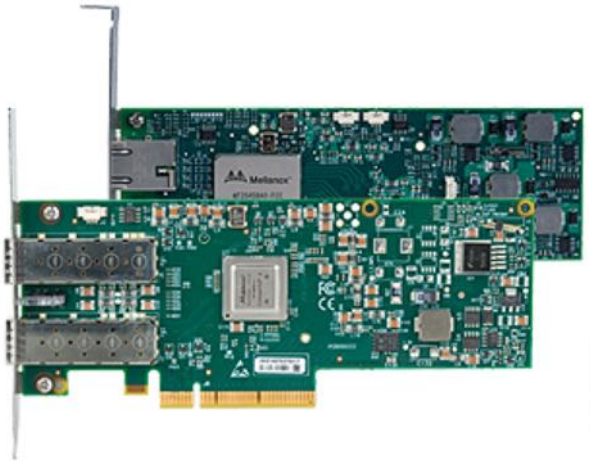
bit/second transmitted

within the computer:

- CPU-Memory: thousand of Mb/sec GByte/sec
  - 10 - 100 Gbit
- CPU- Disks : MByte/sec
  - 50 ~ 100 MB up 1000MB/sec

among computers: networks

- default (commodity)
  - 1000Mbit=1Gbit
- custom(high speed)
  - 10Gb and now 100 Gb (and even more)

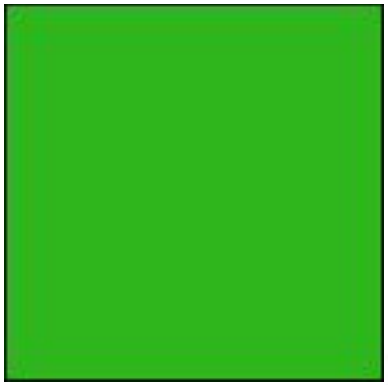


# IOPS vs FLOPS

- HPC is too compute-centric
- Modern Scientific&technical computing requires access to data and computing

**computing 1 calculation**
**≈ 1 picojoule**

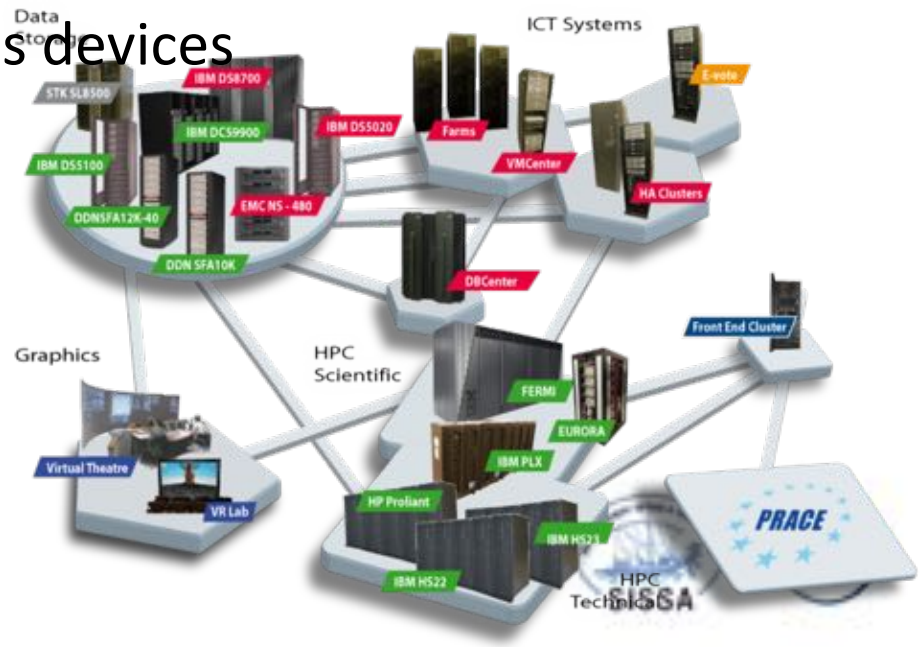
**moving 1 calculation**
**≈ 100 picojoule**



Source: IDC Direction 2013

# Storage size: bytes

- size of storage devices:
  - kbyte/Mbyte -->caches/RAM
  - Gigabyte ---> RAM/hard disks(small size)
  - Terabyte ---> Disks/SAN
  - Petabyte ----> SAN / Tapes devices



MHPC
Master in High Performance Computing

# HPC

# =

# Parallel computing

SISSA ICTP

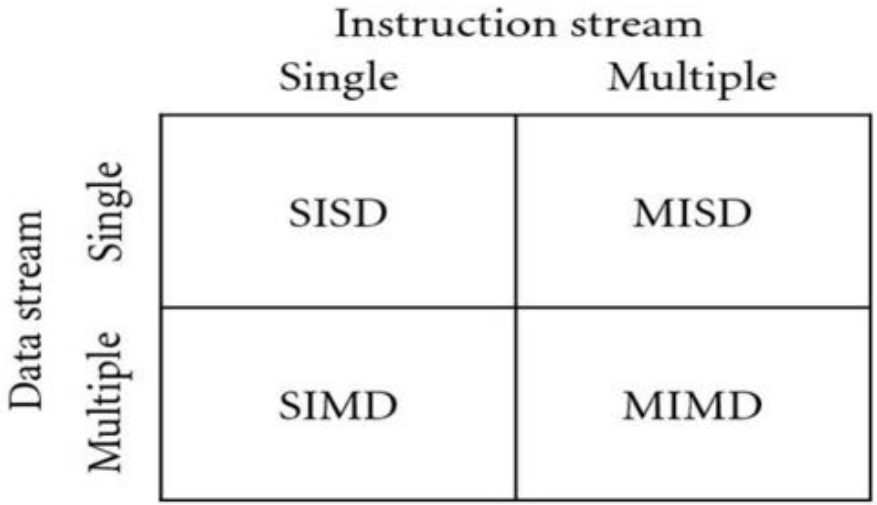
MHPC
Master in High Performance Computing

# PARALLEL HARDWARE

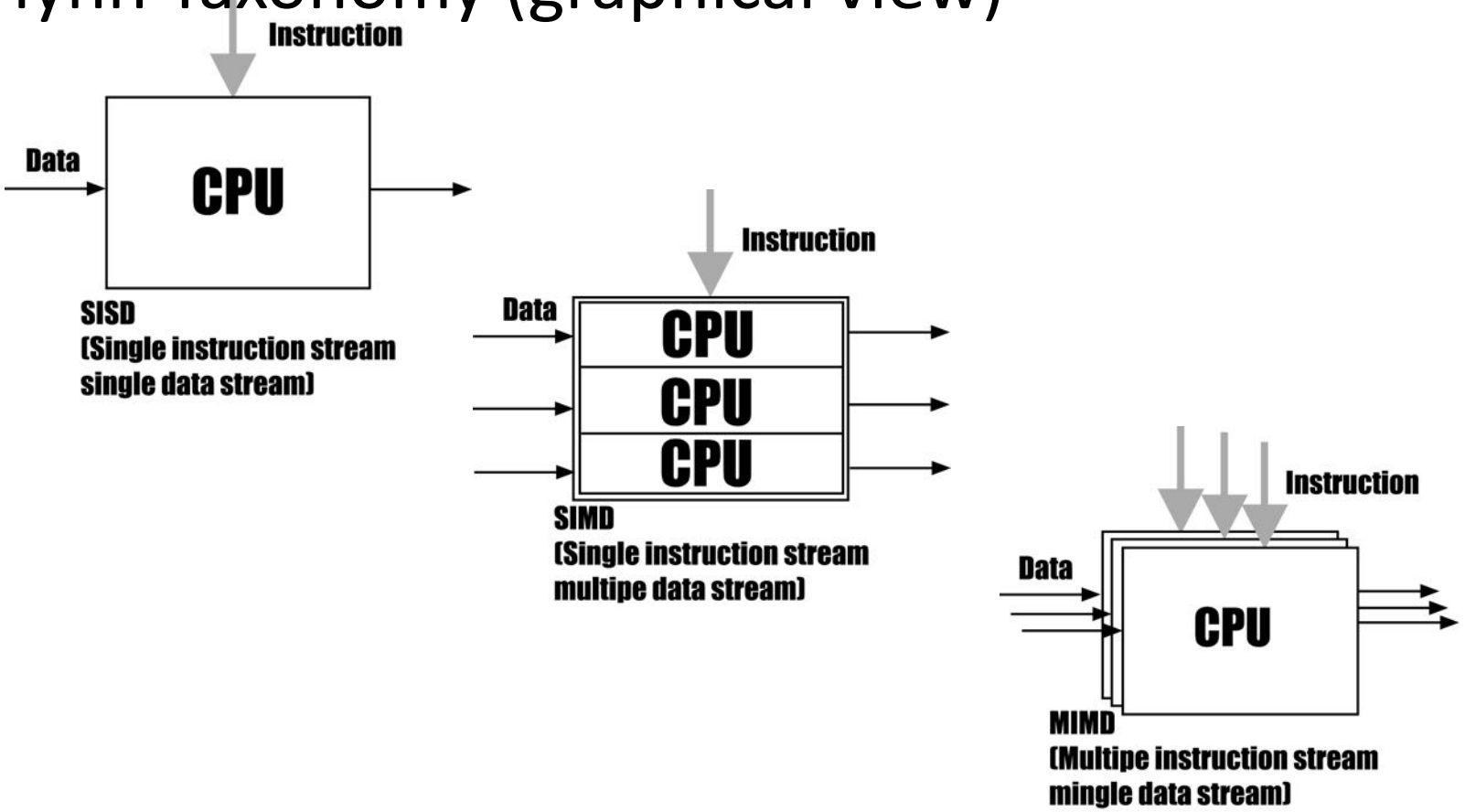
SISSA
ICTP

# Parallel computers

- Tons of different machines !
- Flynn Taxonomy (1966): helps (?) us in classifying them:
  - Data Stream
  - Instruction Stream



# Flynn Taxonomy (graphical view)



MHPC
Master in High Performance Computing

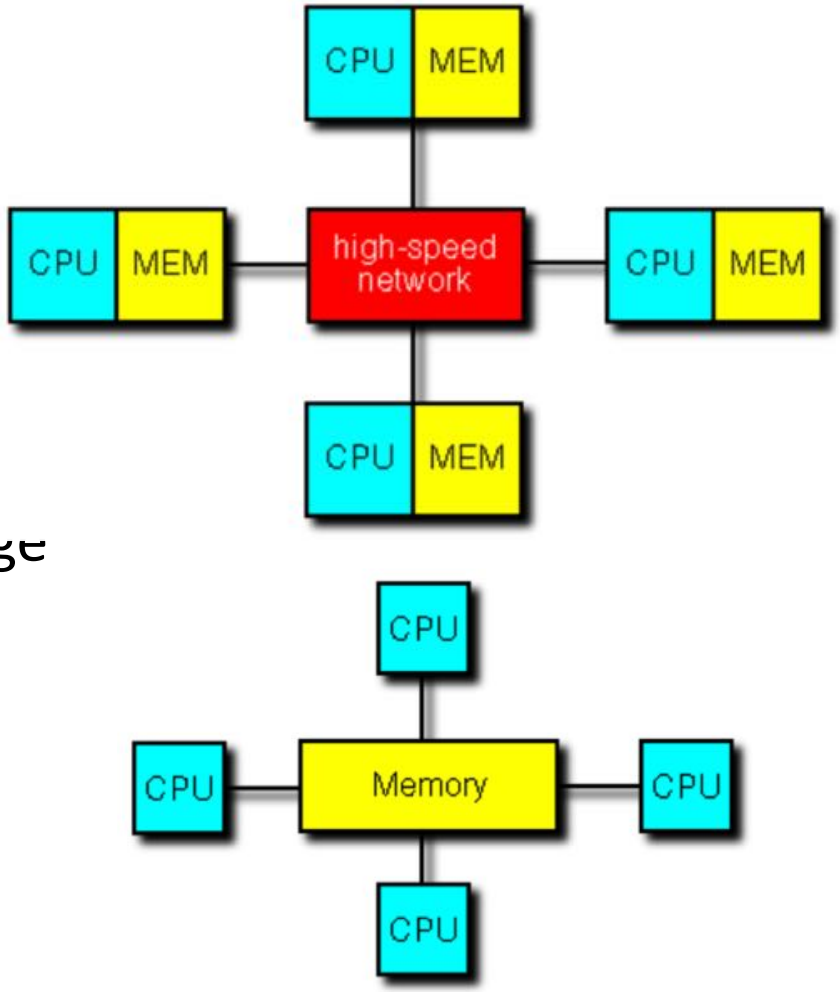
# Another important question:

- MEMORY: The simplest and most useful way to classify modern parallel computers is by their memory model:
  - SHARED MEMORY
  - DISTRIBUTED MEMORY
  - MIXED SITUATION

SISSA ICTP

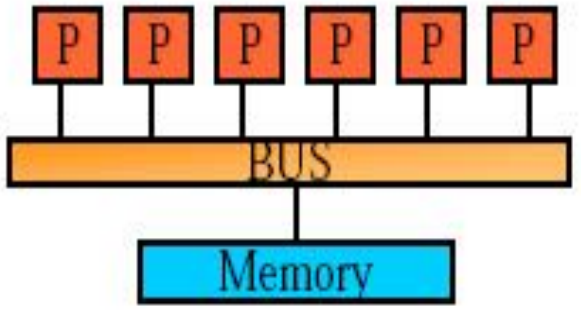
# Shared vs Distributed me

- Distributed memory
  - each processor has its own local memory. Must do message passing to exchange data between processors
- Shared Memory
  - single address space. All processors have access to a pool of shared memory.

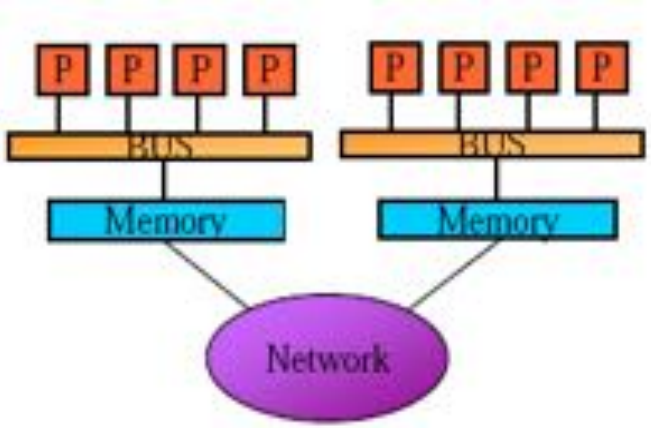


# Shared memory: UMA vs NUMA

*Uniform memory access (UMA):* Each processor has uniform access to memory. Also known as symmetric multiprocessors (SMP)




*Non-uniform memory access (NUMA):* Time for memory access depends on location of data. Local access is faster than non-local access.

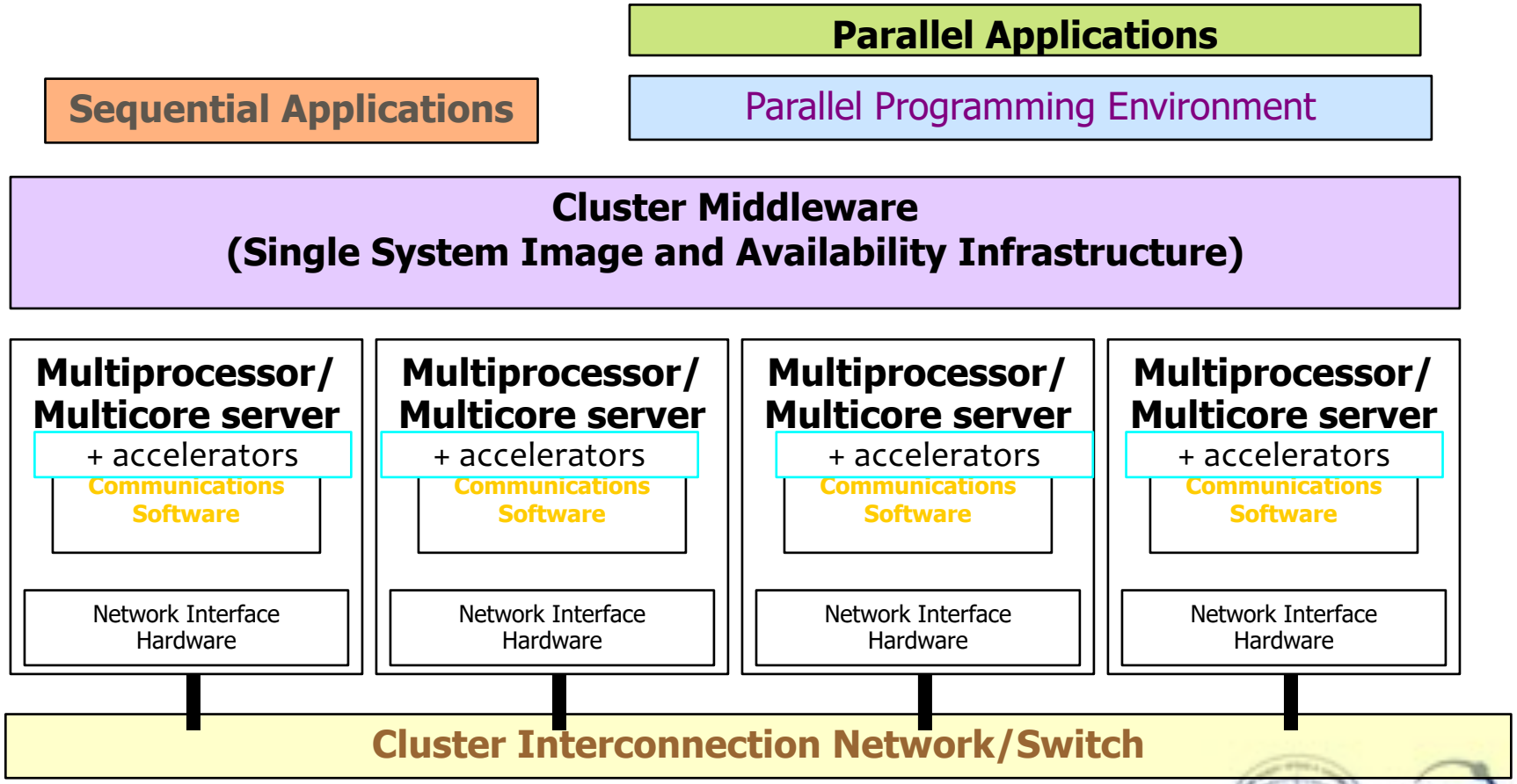


# More on distributed memory machines

- The memory is physically distributed among the processors (local memory). Each processor can access directly only to its own local memory

  NO-Remote Memory Access (NORMA) model
- Communication among different processors occurs via a specific communication protocol (message passing).
- In general distributed memory systems can scale-up from a small number of processors $O(10^2)$ to huge numbers of processors $O(10^7)$
- The performance of the system are influenced by:
  - Features of the node (RAM/cores/CPU frequency/ accelerator)
  - Features (Topology and other) of the interconnection network

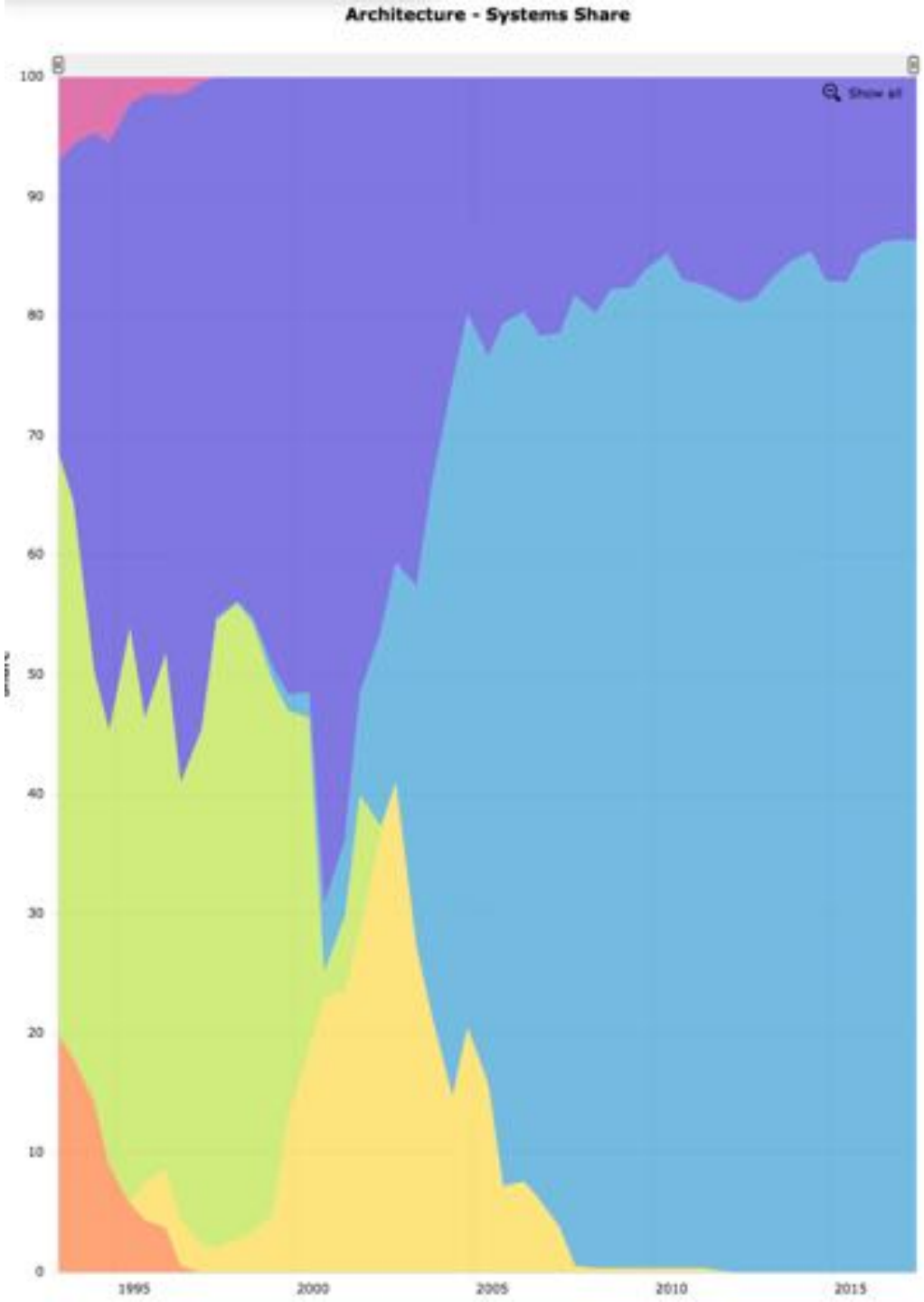
# HPC Cluster Computer Architecture



MHPC
Master in High Performance Computing

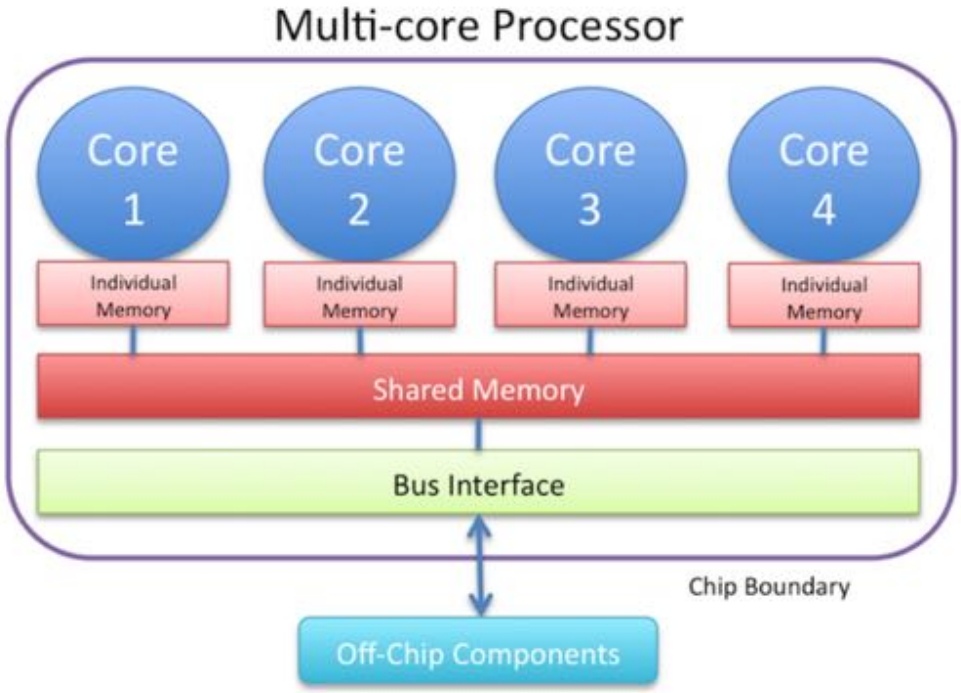
# Top500 supercomputer

- 432 CLUSTERS
- 68 MPP (Massive Parallel Processors)



# Modern CPU are multicore

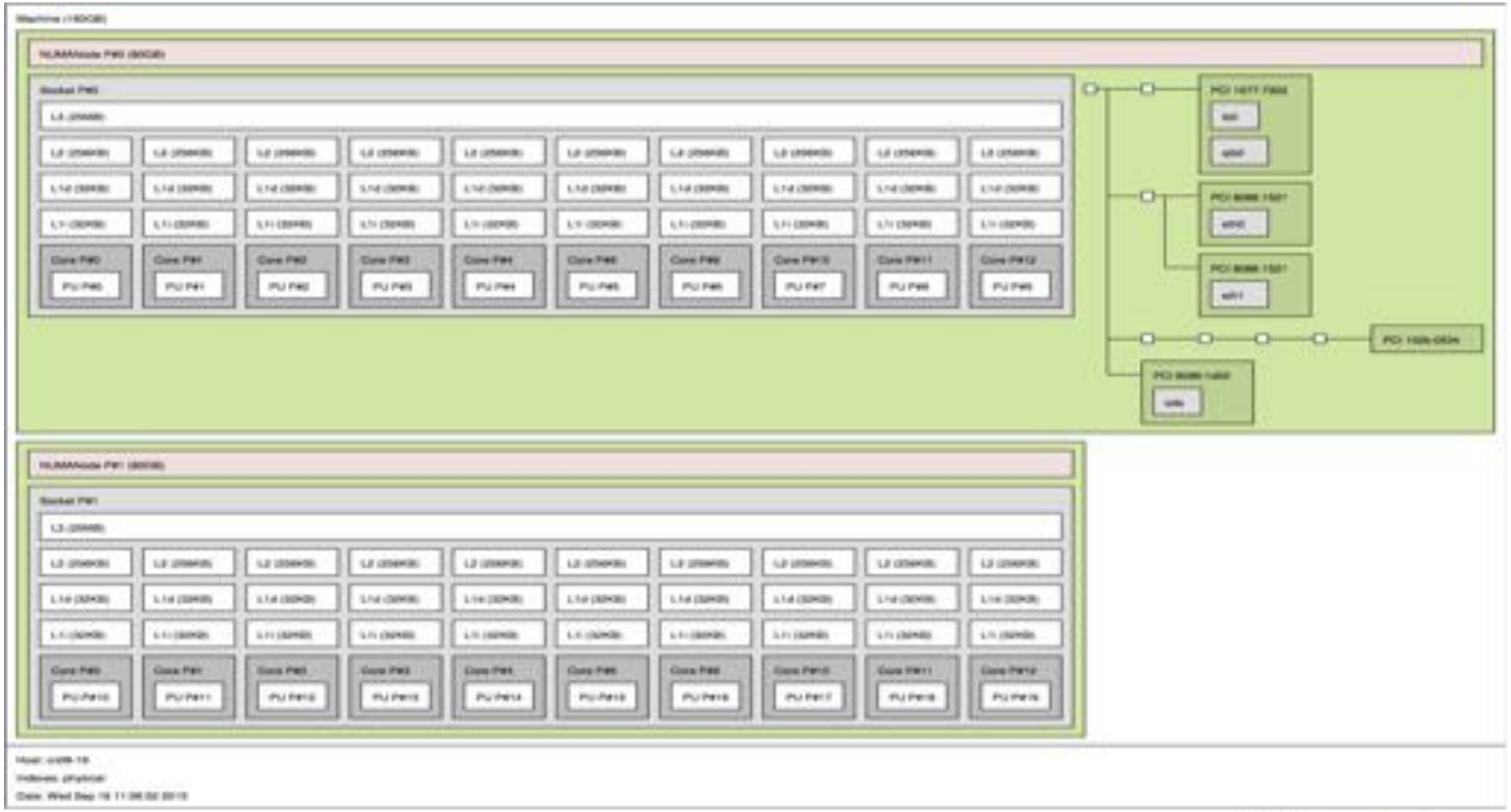
- Because of power, heat dissipation, etc increasing tendency to actually lower clock frequency but pack more computing cores onto a chip.
- These cores will share some resources, e.g. memory, network, disk, etc but are still capable of independent calculations




Picture from: http://www.cse.wustl.edu/~jain/cse567-11/ftp/multcore/

MHPC
Master in High Performance Computing

# A modern node picture (Ullysses node)



MHPC
Master in High Performance Computing

## Parallel Programming Paradigms

Two basic schemes for parallel programming dictated by hardware:

**Shared memory**

Single memory view, all processes (usually threads) could **directly**access the whole memory

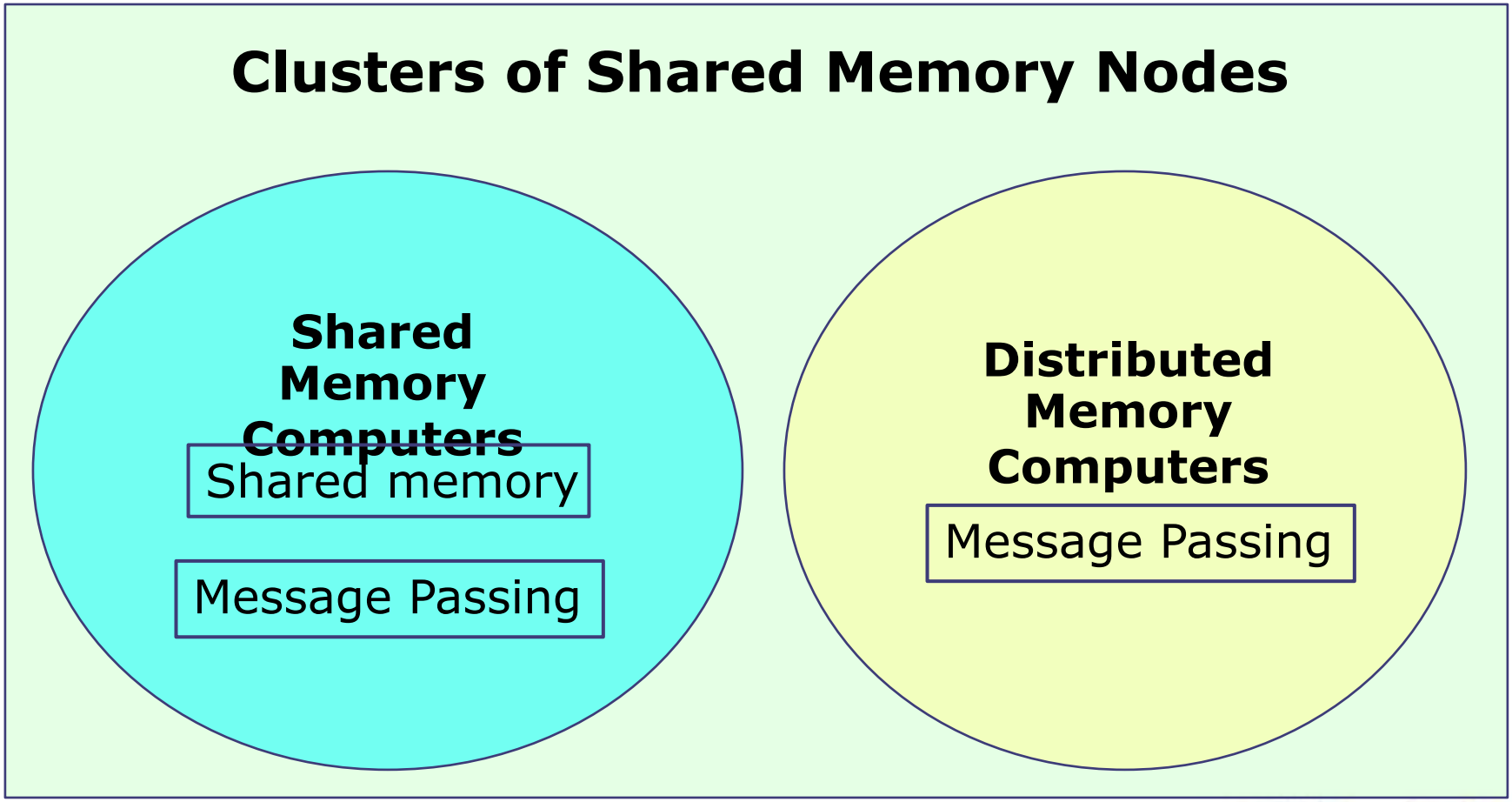
**Message Passing** (distributed memory)

all processes could **directly** access only their local memory

SISSA ICTP

MHPC
Master in High Performance Computing

# Architectures vs. Programming Paradigms




SISSA ICTP

MHPC
Master in High Performance Computing

# How to program shared memory machine ?

- Automatic (implicit) parallelization:
  - compilers do (?) the job for you
- Manual parallelization:
  - Insert parallel directives by yourself to help compilers
  - OpenMP THE standard
- Multi threading programming:
  - more complex but more efficient
  - use a threads library to create task by yourself
- Use already threaded libraries..

SISSA ICTP

MHPC
Master in High Performance Computing

# How to program using message passing ?

- Using the de-facto standard : MPI message passing interface
  - A standard which defines how to send/receive message from a different processes
- Many different implementation
  - OpenMPI
  - Intel-MPI
  - They all provide a library which provide all communication routines
- To compile your code you have to link against a library
  - Generally a wrapper is provided (mpif90/mpicc)

SISSA ICTP

# Other paradigm are now available

Mixed/hybrid approach..

- MPI + OpenMP

Specific SDK for specific devices

- CUDA for Nvdia GPU

Write once run everywhere:

- OpenCL
- OpenACC:
  - OpenACC is about giving programmers a set of tools to port their codes to new heterogeneous system without having to rewrite the codes in proprietary languages.

# Principle of parallel computing

- Speedup, efficiency, and Amdahl's Law
- Finding and exploiting parallelism
- Finding and exploiting data locality
- Load balancing
- Coordination and synchronization
- Performance modeling

All of these things make parallel programming more difficult than sequential programming.

# Speed up

The *speedup* of a parallel application is

Speedup(p) = Time(1)/Time(p)

where

Time(1) = execution time for a single processor

Time(p) = execution time using p parallel processor

If Speedup(p) = p we have *perfect speedup* (also called *linear scaling*)

speedup compares an application with itself on one and on p processors

more useful to compare

The execution time of the best serial application on 1 processor

versus

The execution time of best parallel algorithm on p processors

# Efficiency

The *parallel efficiency* of an application is defined as

$$\text{Efficiency}(p) = \text{Speedup}(p)/p$$

- Efficiency(p) <= 1
- For perfect speedup Efficiency (p) = 1

We will rarely have perfect speedup.

- Lack of perfect parallelism in the application or algorithm
- Imperfect load balancing (some processors have more work)
- Cost of communication
- Cost of contention for resources, e.g., memory bus, I/O
- Synchronization time

Understanding why an application is not scaling linearly will help finding ways improving the applications performance on parallel computers.

# Superlinear speedup

Question: can we find "*superlinear*" speedup, that is

$$\text{Speedup}(p) > p \quad ?$$

Choosing a bad "baseline" for T(1) WRONG !!!

Old serial code has not been updated with optimizations

Shrinking the problem size per processor GOOD

- May allow it to fit in small fast memory (cache)

# Amdahl's law

Suppose only part of an application runs in parallel

- Let s be the fraction of work done serially,
- So (1-s) is fraction done in parallel
- What is the maximum speedup for P processors?

Speedup(p) = T(1)/T(p)

T(p) = (1-s)*T(1)/p +s*T(1)
T(p) = T(1)*((1-s) + p*s)/p

assumes perfect speedup for parallel part

$$Speedup(p) = p/(1 + (p-1)*s)$$

Even if the parallel part speeds up perfectly,
we may be limited by the sequential portion of code.

MHPC
Master in High Performance Computing

# Amdahal's law

Which fraction of serial code  is it allowed ?

| > | 2 | 4 | 8 | 32 | 64 | 256 | 512 | 1024 |
|---|---|---|---|---|---|---|---|---|
| 5% | 1.91 | 3.48 | 5.93 | 12.55 | 15.42 | 18.62 | 19.28 | 19.63 |
| 2% | 1.94 | 3.67 | 6.61 | 16.58 | 22.15 | 29.60 | 31.35 | 32.31 |
| 1% | 1.99 | 3.88 | 7.48 | 24.43 | 39.29 | 72.11 | 83.80 | 91.18 |

What about Scalability ???

SISSA ICTP

MHPC
Master in High Performance Computing

# Scaling…

- Scaling  or scalability:  some sort of ration between the performance and the “size” of the HPC infrastructure
  - Usual way to measure size: # of processors
- The ability for some application to increases  speed when the size of the HPC is increased
- The ability to for some application to solve larger problems when the size of the HPC increases..

SISSA ICTP

# Problem scaling

- Amdahl's Law is relevant only if serial fraction is independent of problem size, which is rarely true
- Fortunately "The proportion of the computations that are sequential (non parallel) normally decreases as the problem size increases " (a.k.a. Gustafon's Law)

MHPC
Master in High Performance Computing

# So What Is Scalability?

- to get N times more work done on N processors
- compute a fixed-size problem N times faster
  - Strong scaling
    - Speedup S = T1 / TN ; linear speedup occurs when S = N
    - Can't achieve it due to Amdahl's Law (no speedup for serial parts)
- compute a problem N times bigger in the same amount of time:
  - Weak scaling
    - Speedup depends on the amount of serial work remaining constant or increasing slowly as the size of the problem grows
    - Assumes amount of communication among processors also remains constant or grows slowly

SISSA ICTP

MHPC
Master in High Performance Computing

# Why weak scaling tends to work better..

Strong scaling: fixed data/problem set; measure speedup with more processors

Ahmdal law

Weak scaling: data/problem set increases with more processors; measure if speed(efficiency) is the same

Gustafson law

FIRST exercise
evaluate strong/weak scalability of a toy parallel code.

SISSA ICTP

MHPC
Master in High Performance Computing

# END
# PART 1

SISSA ICTP