Introduction to solvers and algorithms for CFD and Astrophysics

M. Guarrasi, I. Spisso
SuperComputing Application and Innovation Department
CINECA, Italy

13 November 2017
Casalecchio di Reno
Italy

Table of contents

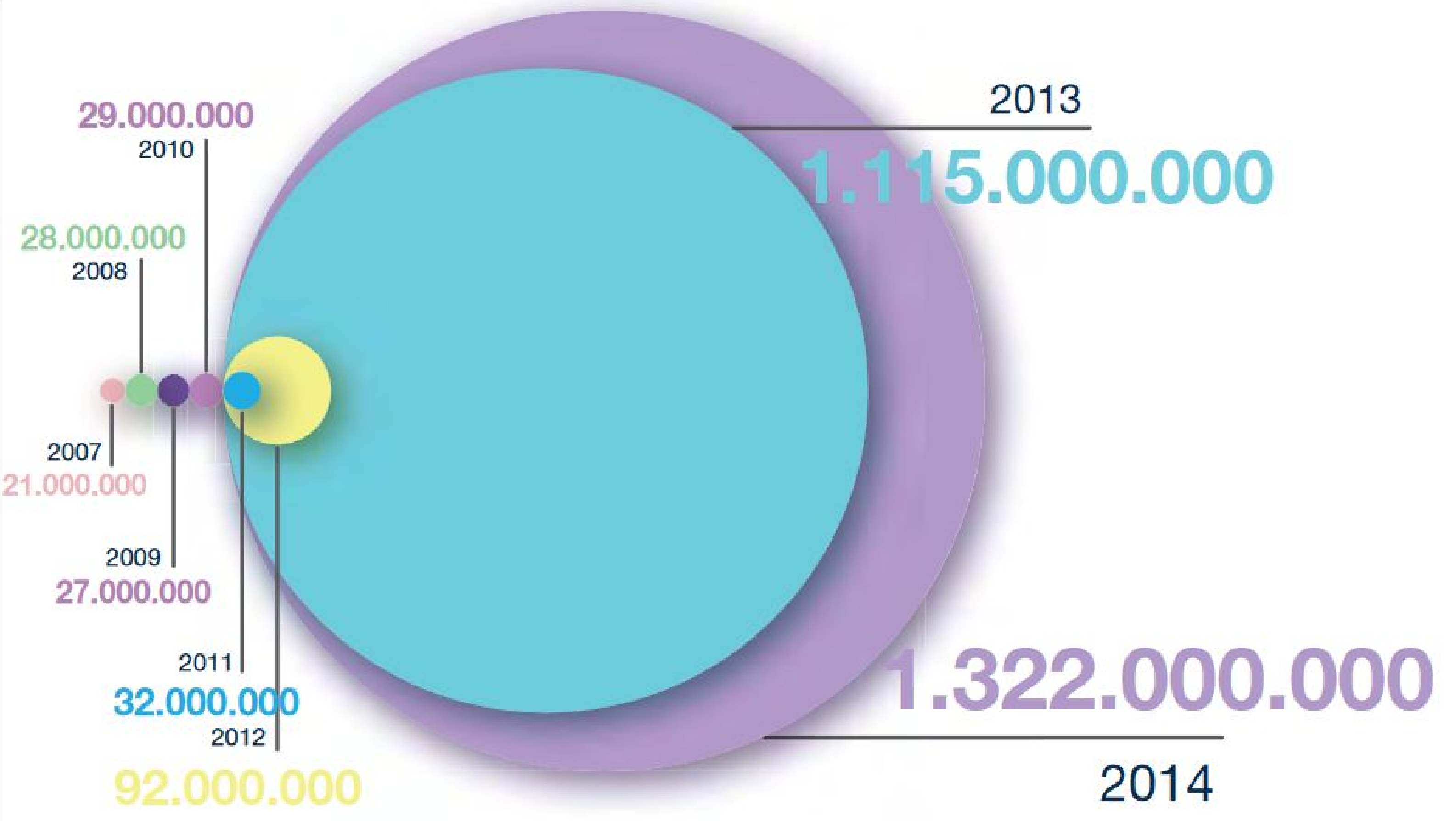
Aim of the workshop

- The aim of this workshop is to present the (most) representative HPC numerical methods used in the fields of Computational Fluid Dynamics (CFD) and Numerical Astrophysics.
- The workshop aims to share the methodologies, numerical methods and their implementation used by the state-of-the-art codes in the HPC environment.
- Key-note lectures will present the challenges of numerically solving Partial Differential Equations (PDE) in problems related to fluid/hydrodynamics, using massively parallel clusters.
- The workshop will focus on state-of the art of the different HPC architecture and the related numerical methods

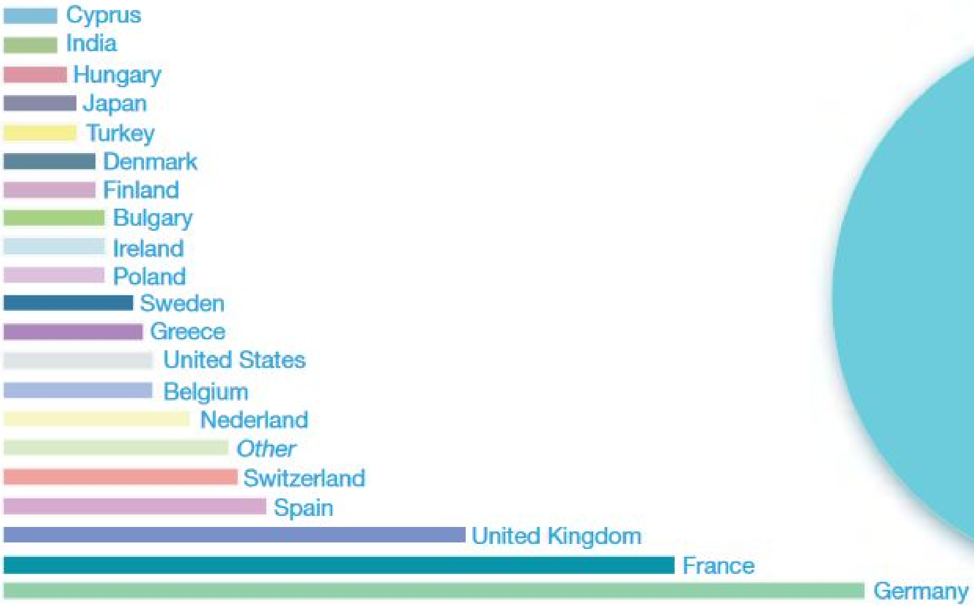
Disclaimer: It is NOT our intent to give a complete survey of the numerical methods used in HPC for the fields of CFD and Numerical Astrophysics. The present workshop shows some of the most used research/community codes granted for access to Tier-0 HPC european (and national) ecosystems in the recent years.

HPC Usage by scientific Sector @ CINECA

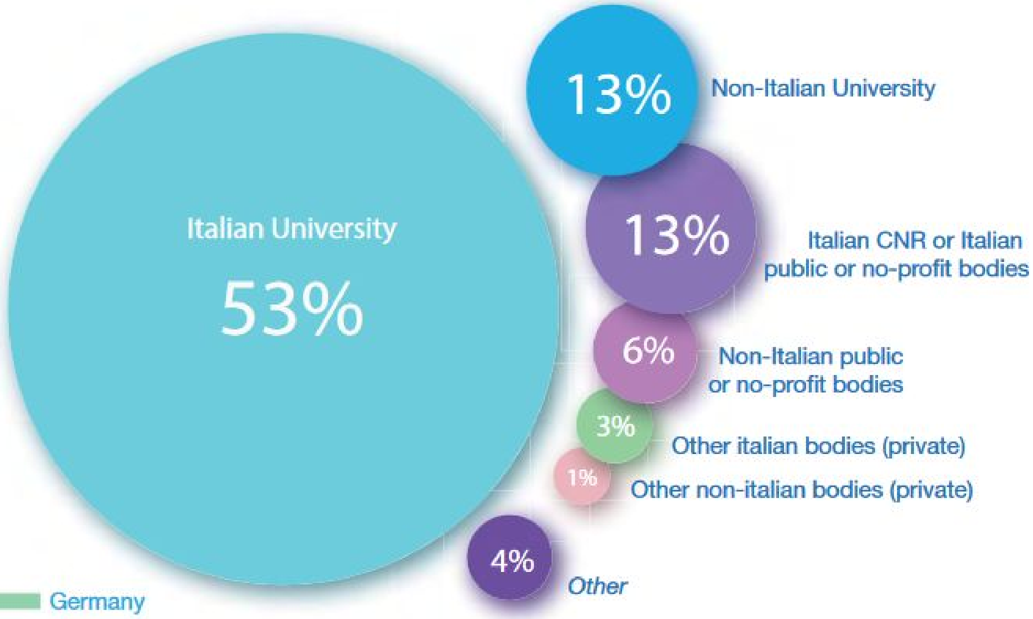
- Scientific computing core hours production



HPC Usage by scientific Sector @ CINECA

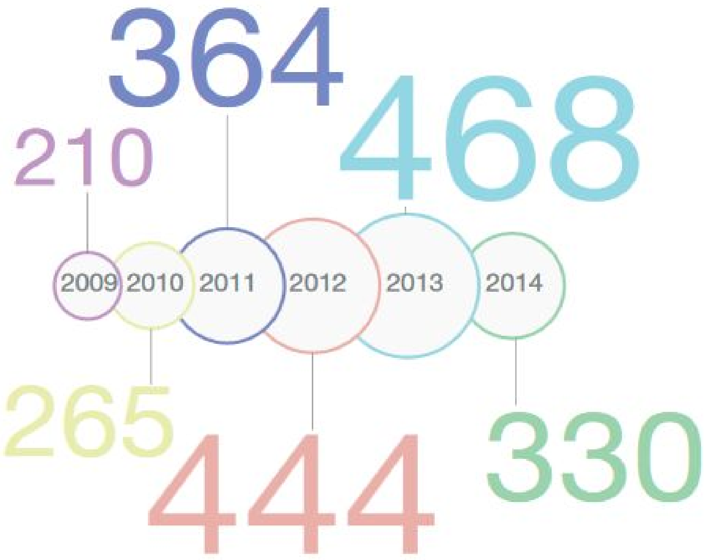


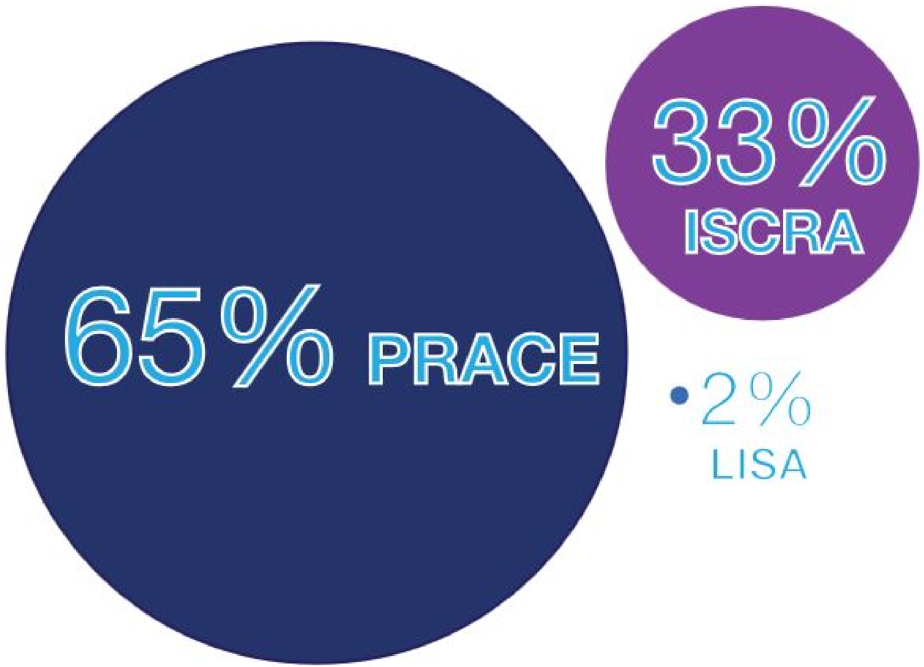
Cineca users with affiliation to foreign entities.



Classification User Institutes

HPC Usage by scientific Sector @ CINECA

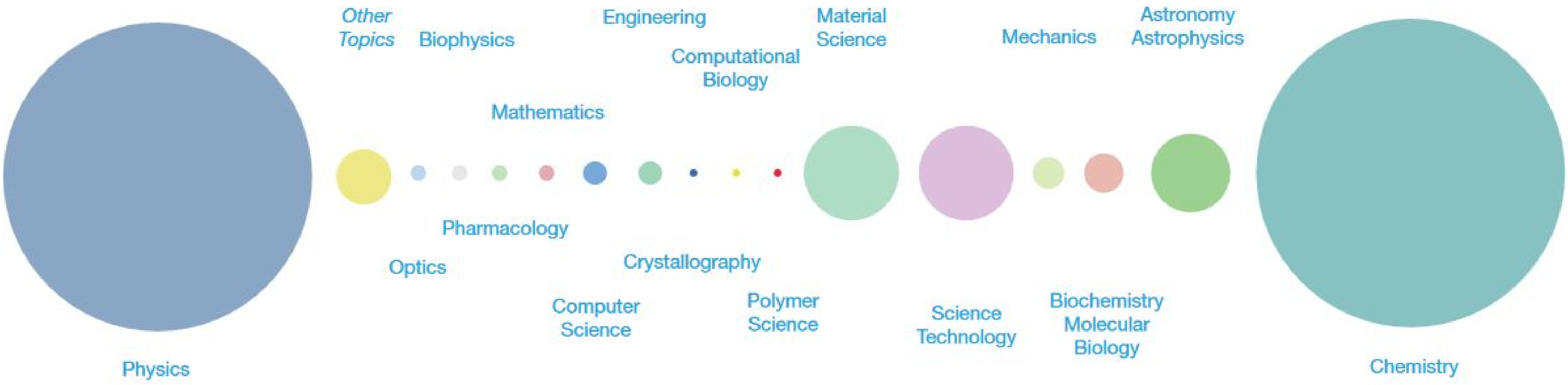




Total number of publications

Allocated resources

HPC Usage by scientific Sector @ CINECA

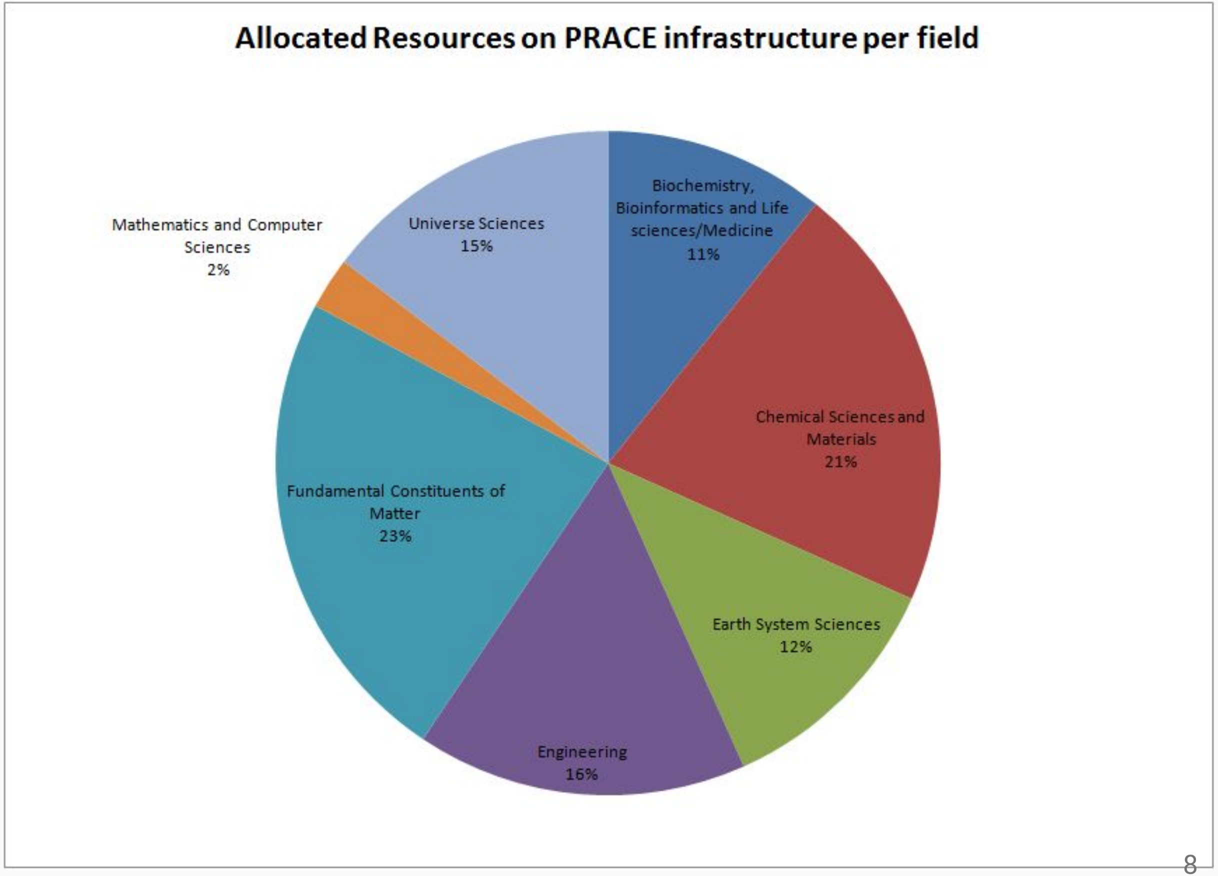


Research areas of the publications mentioning CINECA

HPC Usage by scientific Sector @ PRACE

From Call 1 to Call 13 (2011-now)

- 12 Billions of core hours awarded
- 6 HPC clusters (Now 7)
- 4 hosting members (now 5)
    - CINECA, Italy
    - GENCI@CEA, France
    - BSC, Spain
    - GSC (HLRS, LRZ, JSC), Germany
    - CSCS, Switzerland (starting from call 14)
- 31% ENG + Astro



Overview of Numerical Methods and Algorithms

Layout of following presentations

- Area of research/interest
- Governing Equations
- Numerical Method
- Need for massively parallel clusters
- Implementation in HPC environment and parallelization of the numerical methods
- Use of HPC libraries (if any)
- outcome of HPC grants used (PRACE, ISCRA, etc, etc.)
- future work

Services

Introduction to solvers and algorithms for CFD and Astrophysics

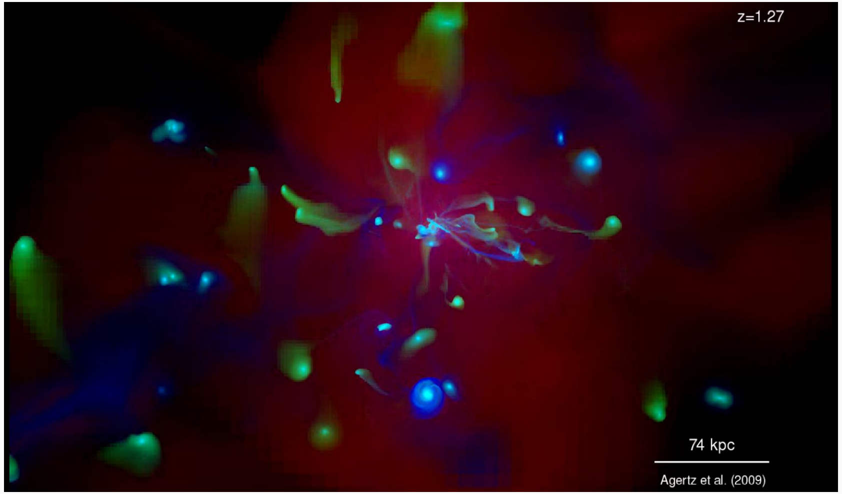
HPC CINECA Infrastructure: State of the art and towards the exascale

Visualizzation of astrophysical and CFD data

Astrophysics

HPC Astrophysical Codes
C. Gheller (CSCS): “The RAMSES codes for computational astrophysics”

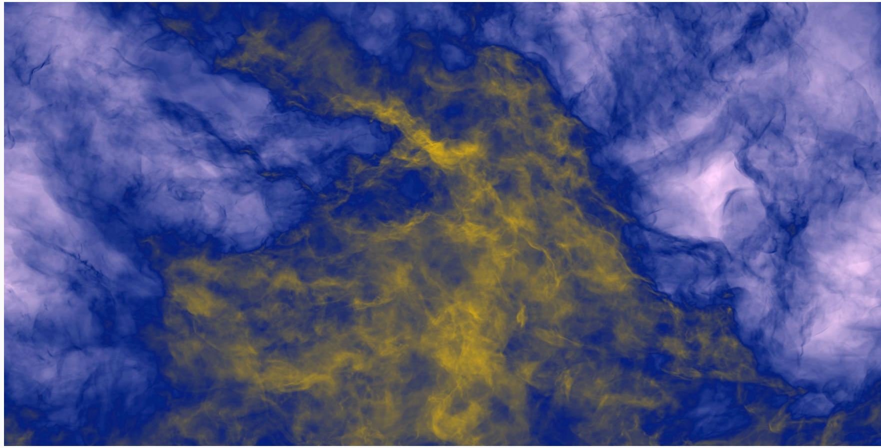
- Name of the codes:RAMSES
- Main Authors: Romain Teyssier (RAMSES)
- Research Area: Cosmology, Galaxy formation, Astrophysics
- Governing Equations: HD, RHD, MHD
- Numerical Method: Various (SPH, PIC, Lagrangian, ….)
- Implementation:
  - Written in F90 and C/C++
  - Parallelization MPI
  - HDF5
  - AMR
- Scalability: excellent weak and strong scaling up to 20-40k



From http://www.ics.uzh.ch

HPC Astrophysical Codes
F.Vazza (UniBO): “Challenges and goals of Eulerian MHD in cosmology”

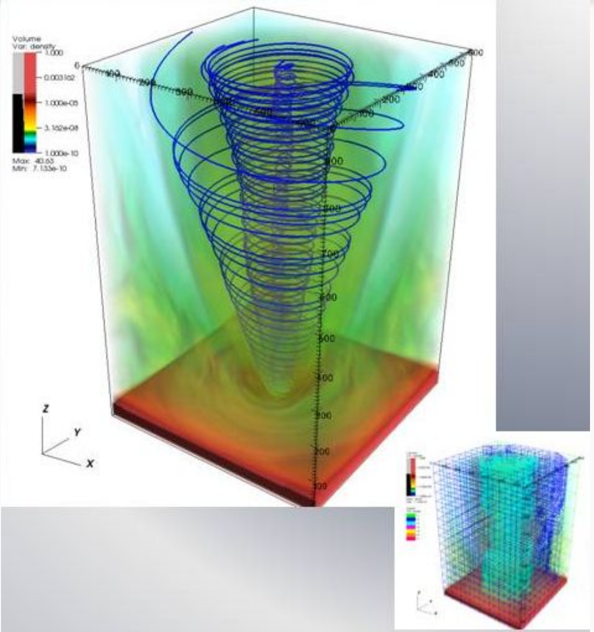
- Name of the codes: ENZO
- Main Authors: Greg Bryan + ENZO collaboration
- Research Area: Cosmology, Galaxy formation, Star formation, Galaxy Clusters
- Governing Equations: HD, MHD, Chemistry, Radiative processes...
- Numerical Method: Various (PPM, PLM, Zeus..)
- Implementation:
  - Written in F90 and C/C++
  - Parallelization MPI
  - HDF5
  - AMR
  - CPU/GPU implementations of HD/MHD
- Scalability: excellent weak and strong scaling up to 20-40k



From http://enzo-project.org/

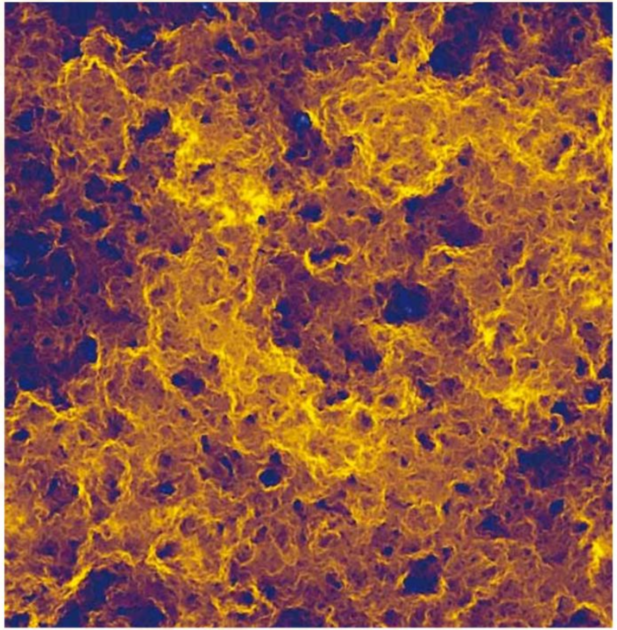
HPC Astrophysics Codes
A. Mignone (UniTO): “The PLUTO Code, an introduction” + “Tutorial on PLUTO”

- Name of the code: PLUTO
- Authors: Mignone, A.; Bodo, G.; Massaglia, S.; Matsakos, T.; Tesileanu, O.; Zanni, C.; Ferrari, A.
- Research Area: Astrophysics/Plasma Physics
- Governing Equations: HD, MHD, RHD, RMHD
- Numerical Method: multi-physics, multi-algorithm modular environment oriented towards the treatment of astrophysical flows in presence of discontinuities
- Implementation:
  - Written in C
  - parallelization pure MPI
  - I/O by HDF5
  - CHOMBO
- Scalability: excellent weak and strong scaling up to 200K cores



HPC Astrophysical Codes
M. Baldi (UniBO): “Numerical methods for standard and non-standard cosmological simulations: The Gadget 3 code”

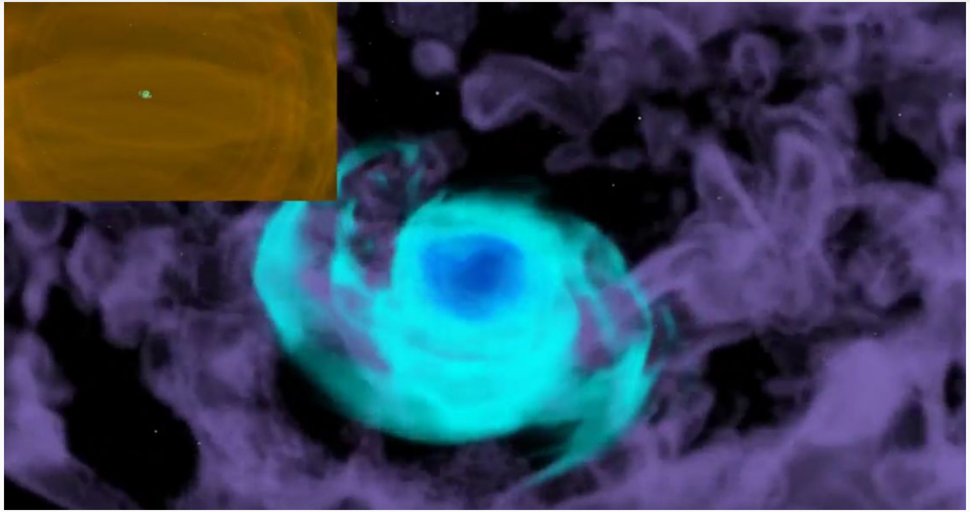
- Name of the code: GADGET 3
- Main Author: Volker Springel
- Research Area: Astrophysics/Cosmology,
- Governing Equations: multi-physics, RHD
- Numerical Method: SPH, Tree-PM
- Implementation:
    - written in C
    - parallelization MPI + OpenMP
    - DFT by FFTW
    - I/O by HDF5
- Scalability: good weak and strong scaling up to 10K-30k cores



© Millenium XXL Project

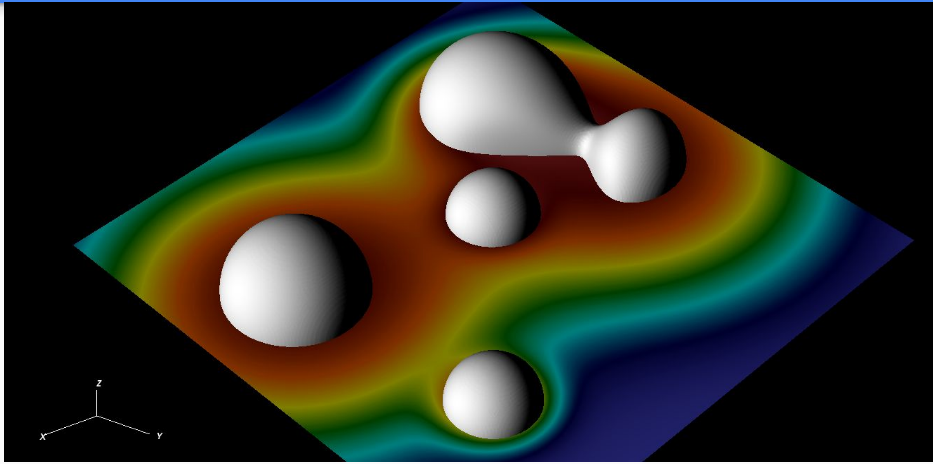
HPC Astrophysical Codes
S. Bernuzzi (UniPR): “Numerical relativity in the gravitational-wave astronomy era”

- Name of the code: BAM
- Main Author/Authors: Bruegmann, B and others
- Research Area: numerical relativty and compact binaries mergers
- Governing Equations: GR + GRHD
- Numerical Method: EULERIAN
- Implementation:
  - Written in C
  - Parallelization: MPI and OpenMP
- Scalability: 8k cores



HPC Astrophysical Codes
B. Giacomazzo: “The Einstein Toolkit: an open framework for Numerical General Relativistic Astrophysics”

- Name of the code: Einstein Toolkit
- Authors: F. Löffler, J. Faber, E. Bentivegna, T. Bode, P. Diener, R. Haas, I. Hinder, B. C. Mundim, C. D. Ott, E. Schnetter, E. Allen, M. Campanelli, and P. Laguna.
- Research Area: Astrophysics, General relativity, Plasma physics
- Governing Equations: GRHD, GRMHD
- Numerical Methods: Various (TVD, PPM, ENO, ePPM, WENO5, MP5, ...)
- Implementation:
  - Written in F90 and C
  - parallelization MP
  - I/O by HDF5
- Scalability: excellent weak and strong scaling up to 10-30k cores



From http://einsteintoolkit.org

HPC CFD code

Useful definiton

In the context of HPC, there are two common notions of scalability:

The first is ***strong scaling***, which is defined as how the solution time varies with the number of processors for a fixed *total* problem size.

The second is ***weak scaling***, which is defined as how the solution time varies with the number of processors for a fixed problem size *per processor*.

HPC CFD Codes: P. Orlandi, A minimal flow unit for turbulence, combustion and astrophysics

- Authors: P. Orlandi, S. Pirozzoli, M. Bernardini
- Research Area: DNS of turbulent low-speed flows. Homogeneous isotropic turbulence, channel and pipe flows (with rotation and roughness elements), passive scalars and inertial particles
- Governing Equations: Incompressible Navier Stokes (DNS)
- Numerical Method: Method-of-lines, two-stage discretization.
  - Spatial discretization on Cartesian staggered grid, Immersed boundary method, second-order FD
  - Time advancement, hybrid third-order Runge-Kutta/Crank-Nicholson scheme
  - Fractional-step: explicit treatment of the convective terms, implicit treatment of the viscous ones
- Implemented in F90, parallelization pure MPI
  - FFTs and tridag systems exploit available libraries (FFTW or IBM ESSL)
- Scalability: excellent weak and strong scaling for channel flow simulations on FERMI

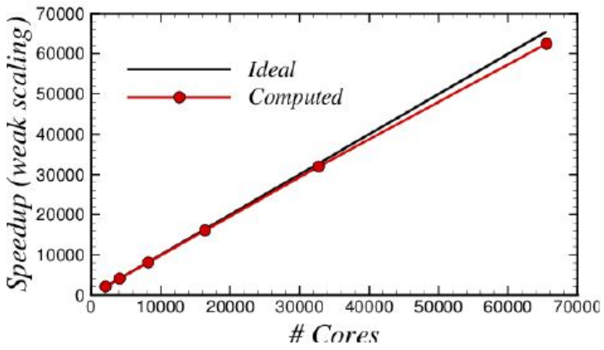


Figure 3: outcome of weak scalability tests

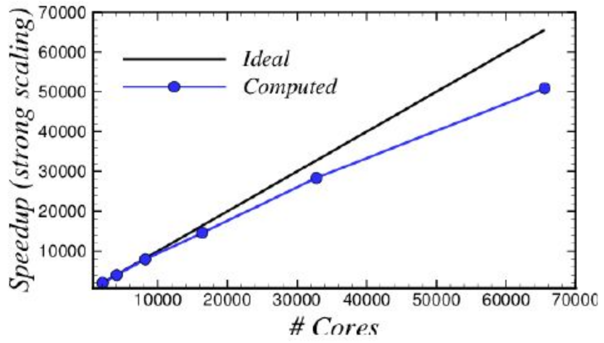
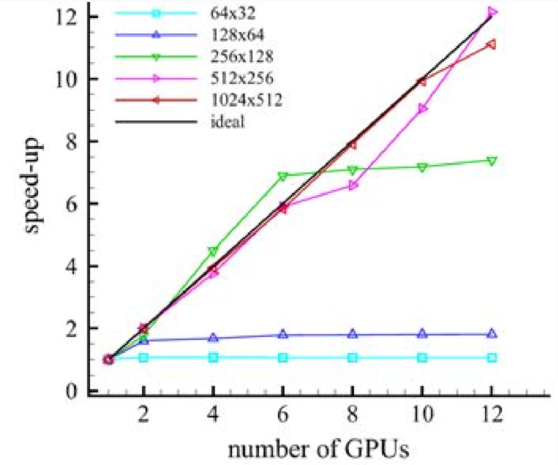
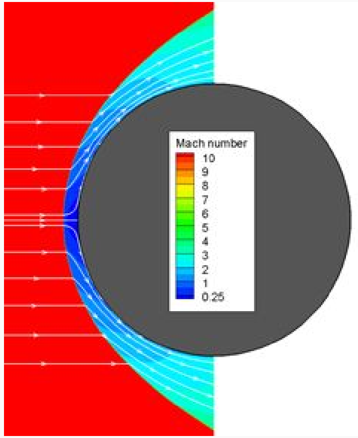


Figure 4: outcome of strong scalability tests.

HPC CFD Codes: F. Bonelli, CFD and state-to-state of hypersonic flows using GPUs

- Authors: Francesco Bonelli, Michele Tuttafesta, Gianpiero Colonna, Luigi Cutrone, Giuseppe Pascazio
- Research Area: hypersonic flows in thermochemical non-equilibrium
- Governing Equations: compressible Navier-Stokes or Euler equations
- Numerical Method: Cell-centered Finite Volume Space discretization on a Multi-block structured mesh, Operator splitting approach, Method-of-lines
  - Space discretization:
    Inviscid flux: Flux Vector Splitting of Steger and Warming or AUSM with MUSCL approach for higher order accuracy;
    Viscous flux: gradients of the primitive variables are evaluated by applying Gauss theorem
  - Time integration: Runge-Kutta scheme up to third order
  - Implemented in CUDA C, parallelization with MPI and CUDA
  - Scalability: excellent strong scaling for mesh larger than 256x128 and excellent weak scaling (computation performed on a GPU cluster available at "Politecnico di Bari")
  - Speed up GPU vs single core CPU: up to 150

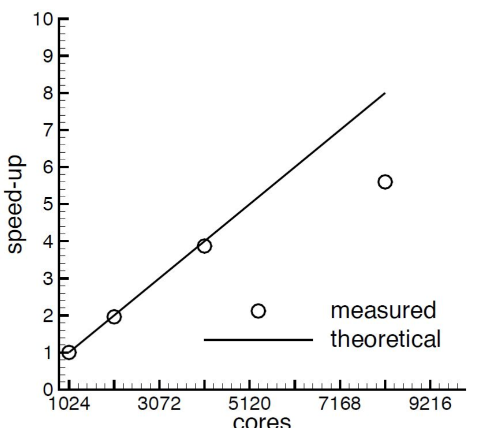


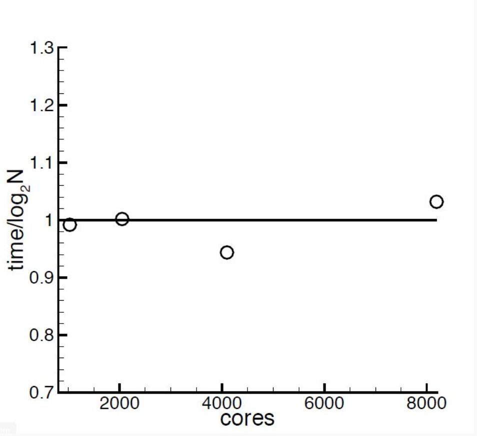


Strong scaling (left) and hypersonic flow over a sphere (right)

HPC CFD codes: P. Gualtieri, Particle-laden turbulent flows: small scale clustering and two-way coupling effects

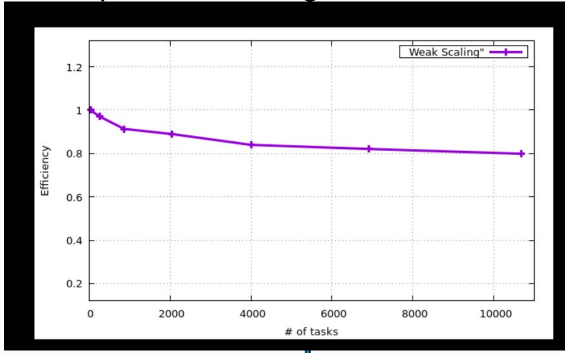
- n_s_trip (Navier Stokes Triperiodic)
- Authors: P. Gualtieri, F. Battista, J.P. Mollicone, C.M. Casciola in collaboration with Giorgio Amati (OpenMP code developed @Caspur) and with Francesco Salvadore (MPI code developed @Cineca)
- Research Area: particle laden turbulent flows, (multi-scale) turbulent transport, two-way coupling effects
- Governing Equations: Incompressible Navier-Stokes equations
- Numerical Method: Method-of-lines
    - Space discretization: Pseudo-spectral
    - Time integration: Low Storage 4th order Runge-Kutta
    - Written in Fortran77 (OpenMP) ; Fortran 90 (MPI)
    - Parallelization: Both OpenMP & MPI
    - Numerical librarires: NCAR and ESSL in the OpenMP version. P3dFFT and FFTW in the MPI version
    - Serial I/O (OpenMP) and MPI-I/O
- Scalability:
    - Strong scaling N=1536^3 Fourier Modes (left
    - Weak scaling (right)

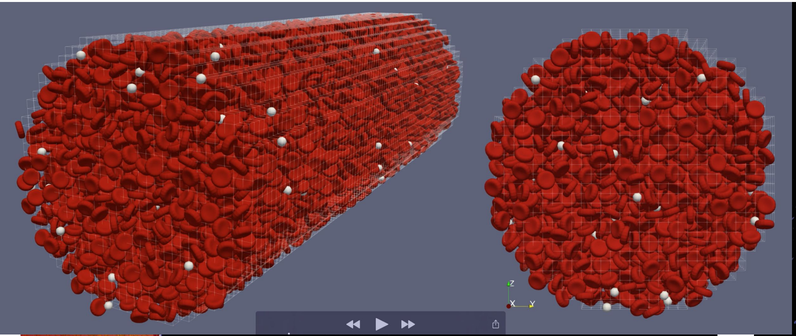


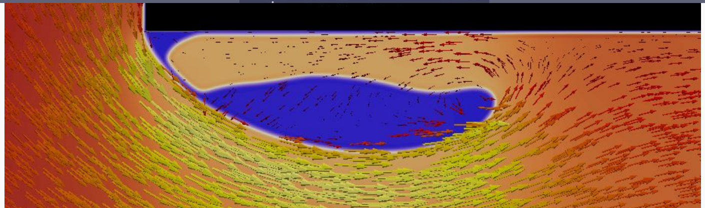


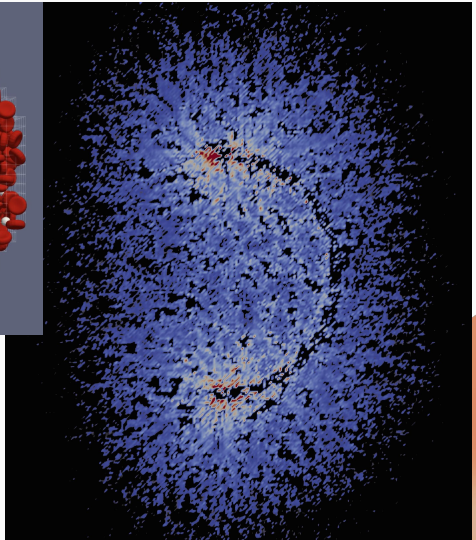
HPC Codes: G. Falcucci: Multi-scale Modeling of complex flows through the Lattice Boltzmann Method

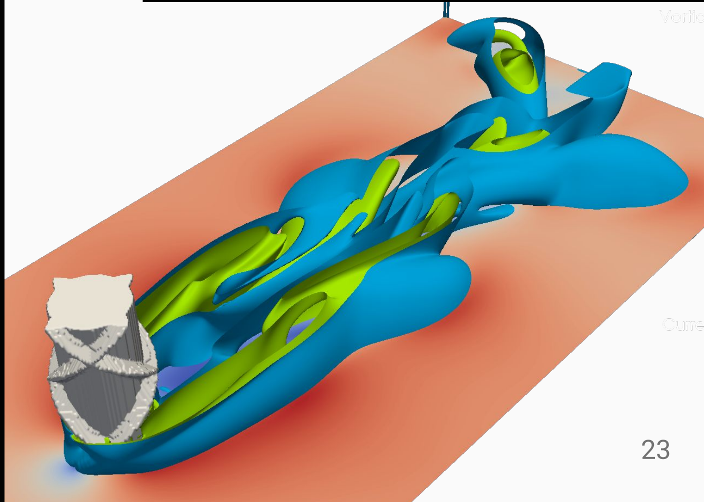
- Authors: G. Falcucci and S. Succi
- Research Area: multi-scale modeling of complex flows, multiphase pseudo-potential, multicomponent/reacting flows, FSI
- Governing Equations: Kinetic Theory
- Numerical Method: Lattice Boltzmann
  - Written in Fortran77 and Fortran90 (+ Python scripting for post-processing)
  - Parallelization: OpenMP and MPI
- Scalability: scalability on hundreds of computational cores (and even more)





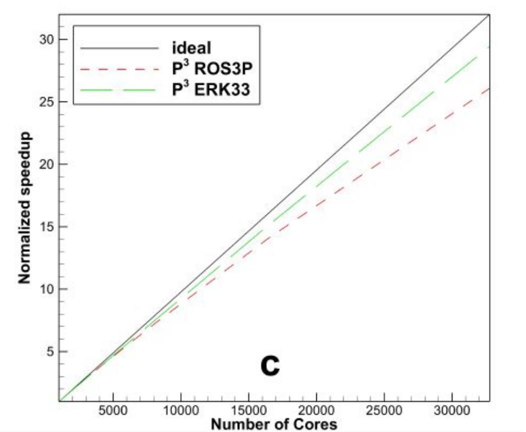






HPC CFD Codes: A. Colombo, Discontinuous Galerkin Methods in HPC

- Authors: F. Bassi, A. Colombo, L. Botti, A. Ghidoni, A. Nigro, A. Crivellini
- Research Area: transonic flows, shock boundary layer interaction (SBLI)
- Governing Equations: from Euler equations to the hybrid RANS-LES approaches, inc. and compressible
- Numerical Method: Discontinuous Galerkin method, MIGALE code
  - The equations of all the implemented flow models are discretized to the same high-order accuracy on hybrid (possibly curved) meshes
  - explicit and implicit high-order (up to order six) time integrators implemented to exploit the high-order discretization both in space and time.
  - based on the SPMD (single process, multiple data) paradigm, MPI paradigm
- Numerical Libraries
  - PETSc library to achieve parallelism
- The scalability of the code MIGALE has been investigated on three different TIER-0 and one TIER-1 facilities: CURIE, HORNET and FERMI
  - Good scalability results for all clusters
  - weak scalability up to 32k cores on FERMI



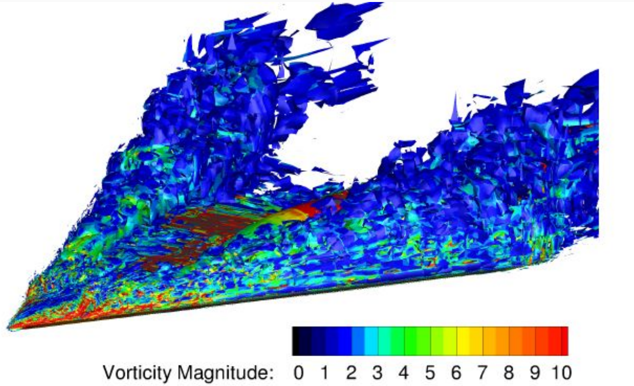


Figure 1: X-LES over the VFE-2 delta wing [1,2]

HPC CFD Codes: OpenFOAM, currente state, perspective and in-situ visualization

- OpenFOAM is become more and more popular in the CFD community
  - OpenFOAM is (aiming to) becoming The open-source community code
    - Third most-used CFD community code by users (after Ansys-Fluent and CD-Adapco-Starccm+), http://www.resolvedanalytics.com/theflux/comparing-popular-cfd-software-packages
    - Fifth most-used CFD code in HPC environment
- Does OpenFOAM can seat in this “round table” of Tier-0 CFD codes?
  - Not yet
- Missing for a “full enabling” on massively parallel clusters (Tier-0 size)
  - Pstream (MPI Library) actually scales reasonably well up to orders of thousands of cores
  - Serial I/O, not MPI
  - Actual sparse matrices storage system (LDU) does not enable any cache-blocking mechanism or efficient vectorization
- Work “done/in progress/ to do” inside the community
  - Modified version of Pstream OpenFOAM available scaling up to order of thousands of cores (done v1606)
  - Implementation of Adios MPI I/O library on-going
  - In-situ visualization with Catalystic
  - CFD4Exascale focuses on the technologies necessary to transition CFD from its current peta-scale performance point towards exascale deployment.

Thank You