Computational Astrophysics and Cosmology

Open Access Springer Journal

CompAC publishes paper on

- **Astronomy, physics and cosmology**
- **Computational and information science**

The combination of these two disciplines leads to a wide range of topics which, from an astronomical point of view covers all scales and a rich palette of statistics, physics and chemistry. Computing is interpreted in the broadest sense and may include hardware, algorithms, software, networking, data management, visualization, modeling, simulation, visualization, high-performance computing and data intensive computing.

Modeling and Observing Dense Stellar Systems with the Astrophysical Multipurpose Software Environment

Simon Portegies Zwart
Sterrewacht Leiden

AMUSE

Sterrewacht Leiden


MODEST2 Amsterdam (2003)

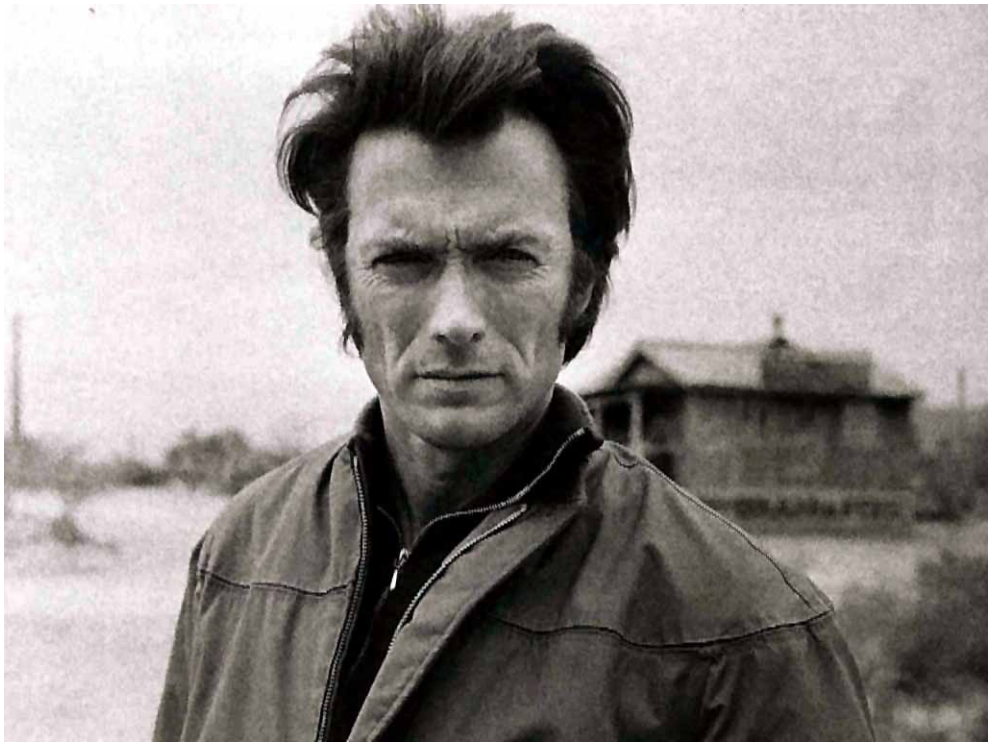
MODEST14 Bad Honnef

AMUSE - philosophy

- Build on community codes
- Standardized interfaces
- Automate as much as possible
- Core Team:
  - Inti Pelupessy (post-doc)
  - Arjen van Elteren (engineer)
  - Nathan de Vries (programmer)
  - David Jansen (user support)

Leiden Dirty Hands Days

- June 18: hydrodynamics AMR, LBM or SPH?
- June 25: Gravity a-la cart, trees, SCF or BS?
- September 17: Radiosity, ray tracing and casting



AMUSE Today

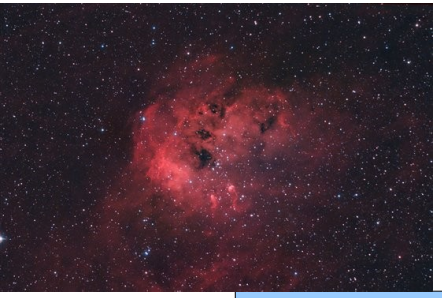
- Automated referencing
- Unit conversion
- Online documentation
- Many of example scripts
- Syllabus and textbook
- Download from: www.amusecode.org

AMUSE's uses

- Astronomical research
  - Multi-scale simulations of astronomical phenomena
  - Multi-physics simulations
- Education at MSc and PhD level
- Public outreach
- Develop a “*gut feeling” of physics*


April 2014

Gravitational dynamics



bridge

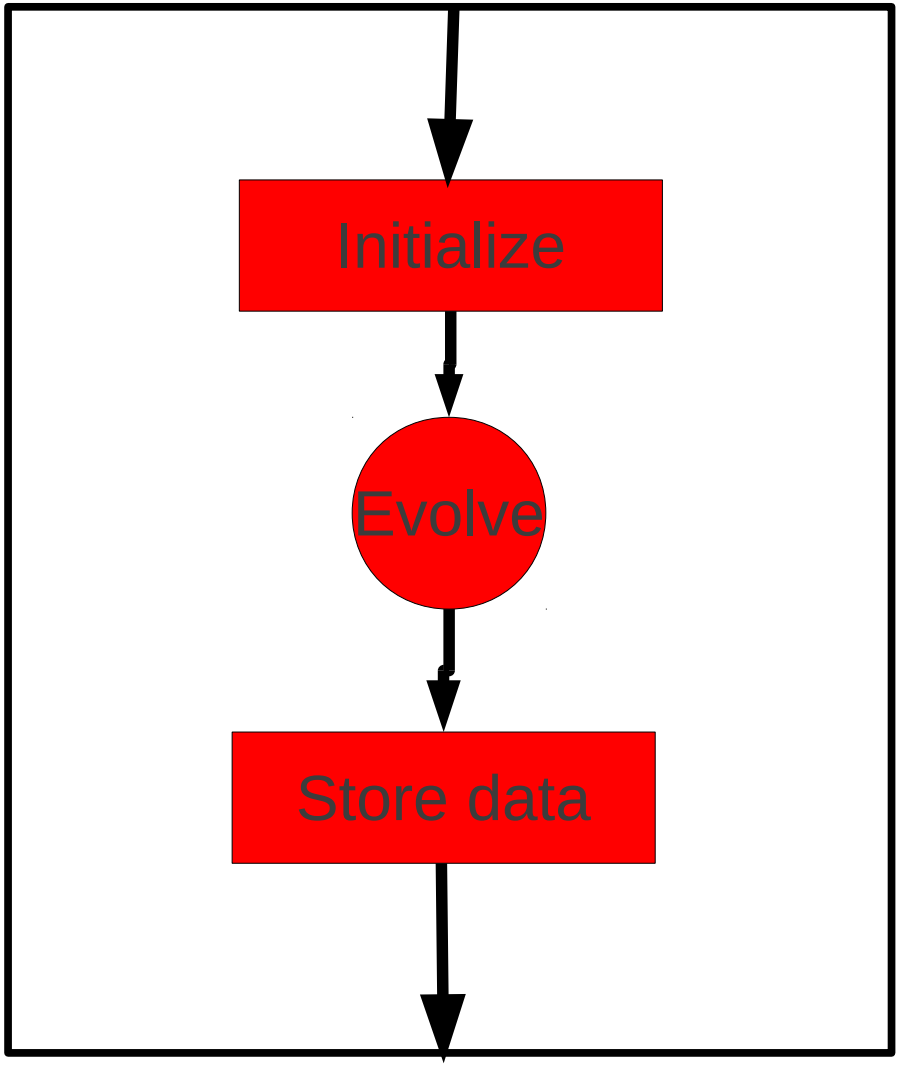
hydrodynamics

AMUSE

Stellar evolution

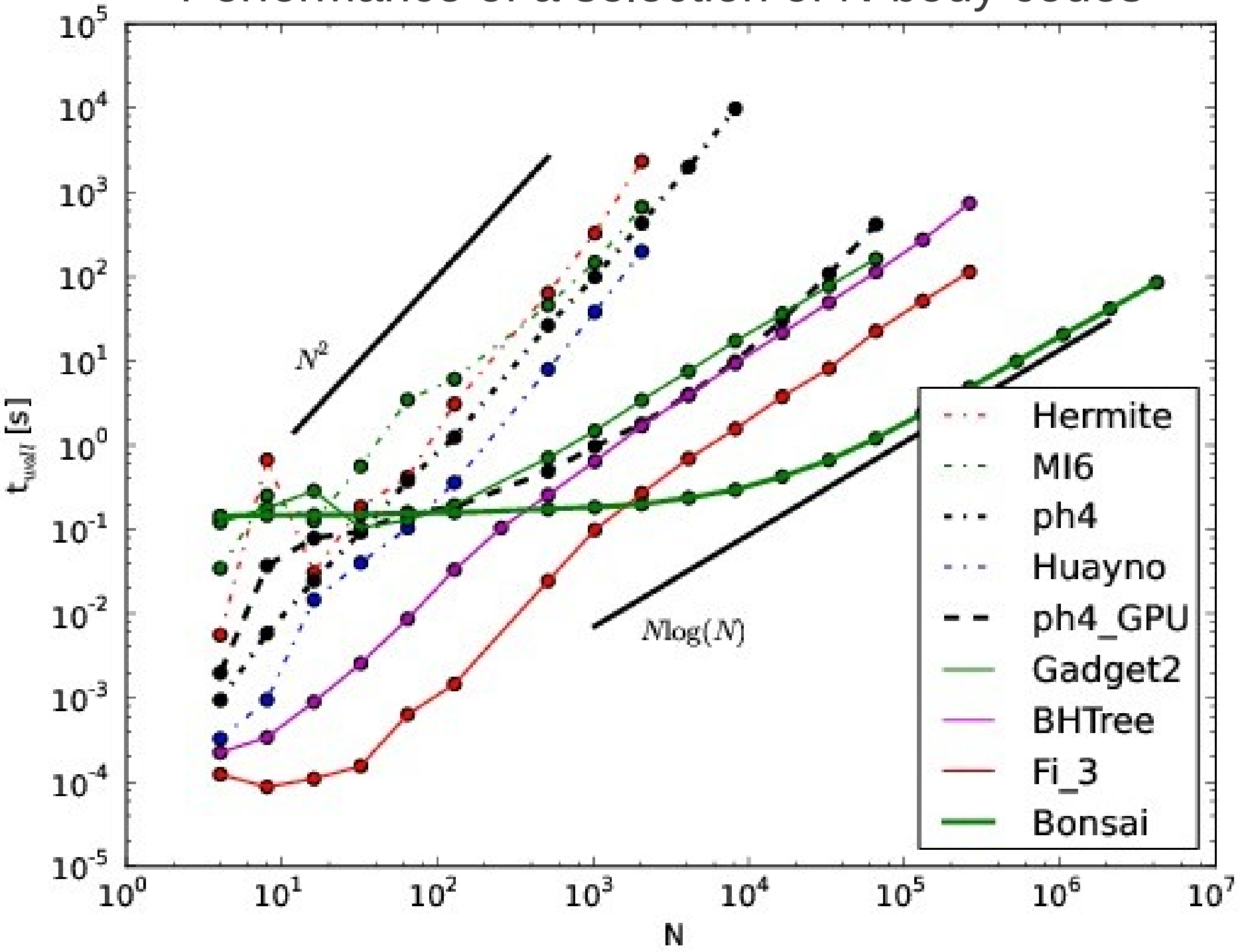
bridge

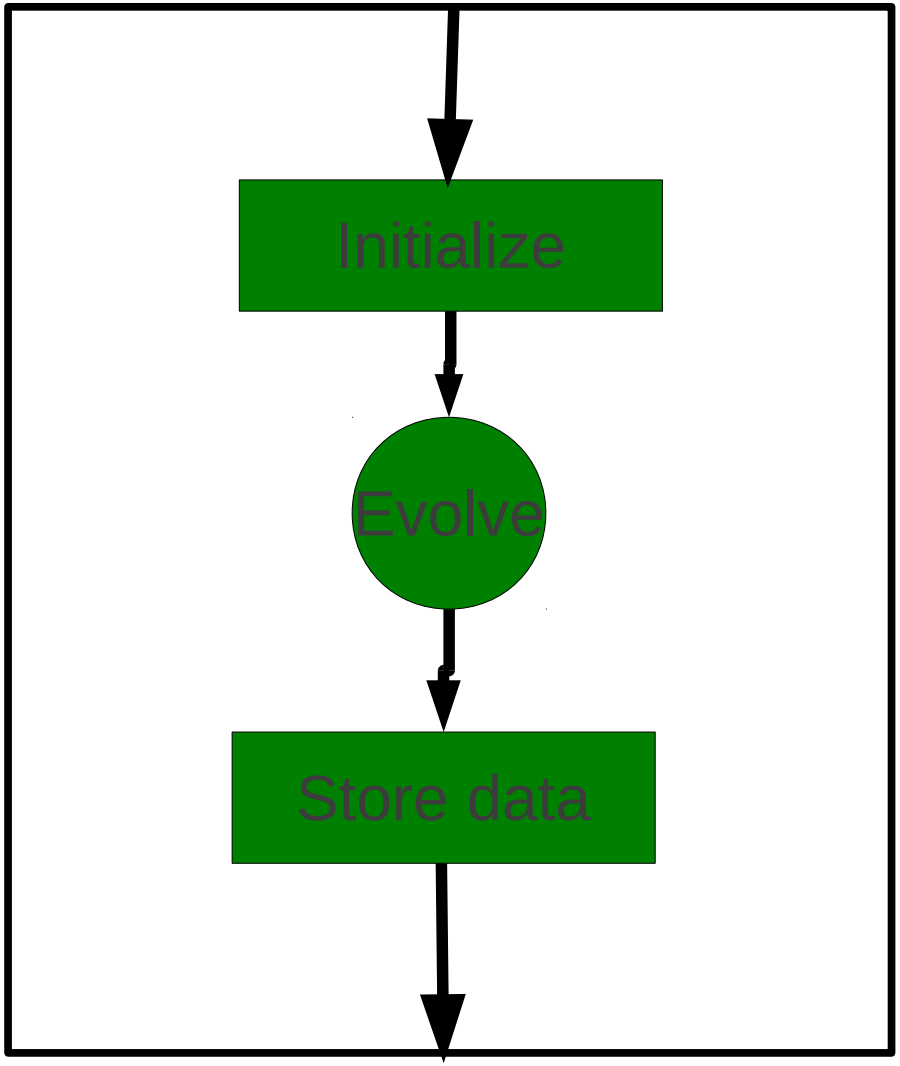
Radiative transport




```
> from amuse.lab import *
> bodies = King_model(N, W0)
> gravity = Hermite()
> gravity.add_particles(bodies)
> gravity.evolve_model(t_end)
>write_to_file(gravity, “grav.hdf5”)
```

Performance of a selection of N-body codes

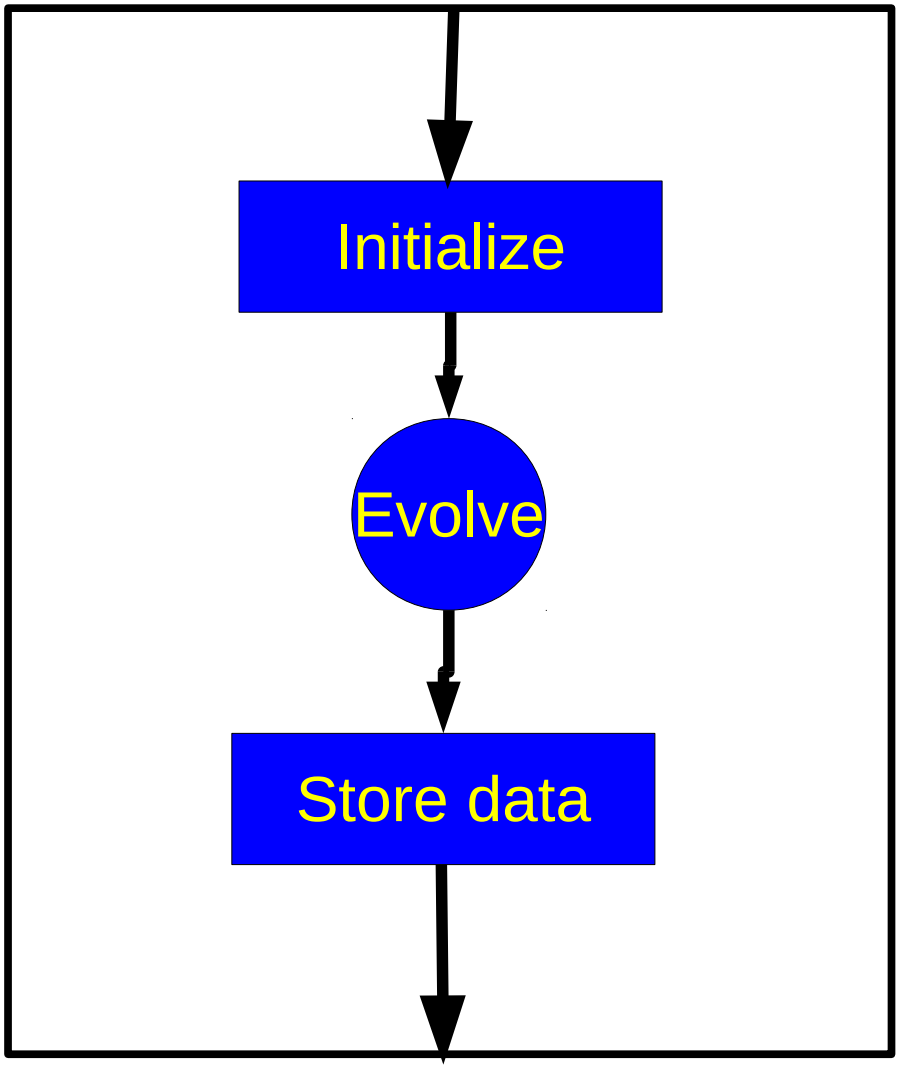




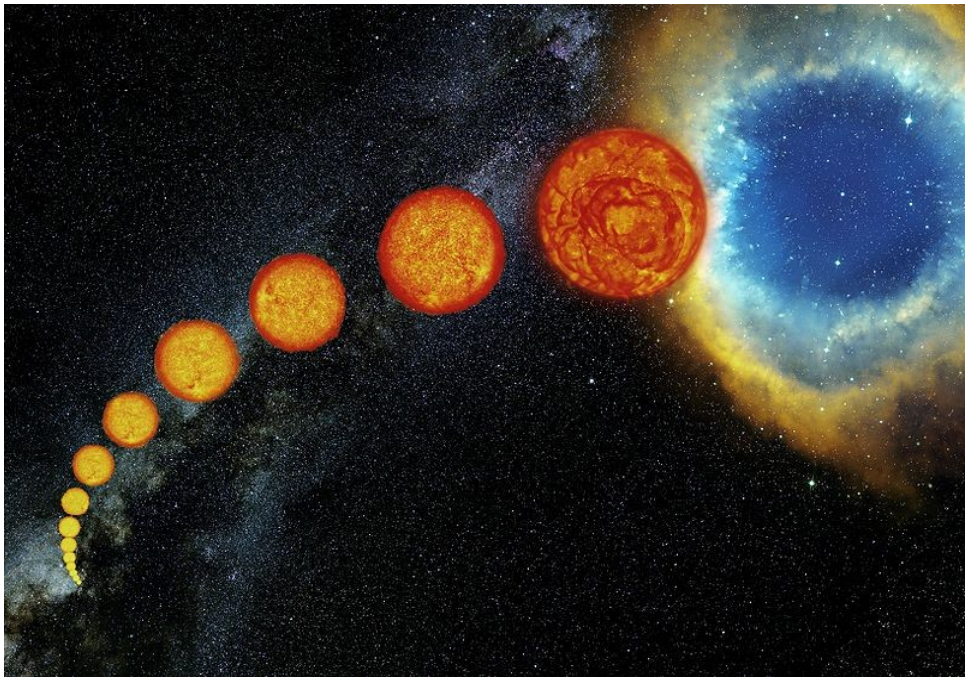

```
> from amuse.lab import *
> bodies = Plummer_gas_model(N)
> hydro = Gadget2()
> hydro.add_particles(bodies)
> hydro.evolve_model(t_end)
> write_to_file(hydro, “hydro.hdf5”)
```

Running in distributed mode:

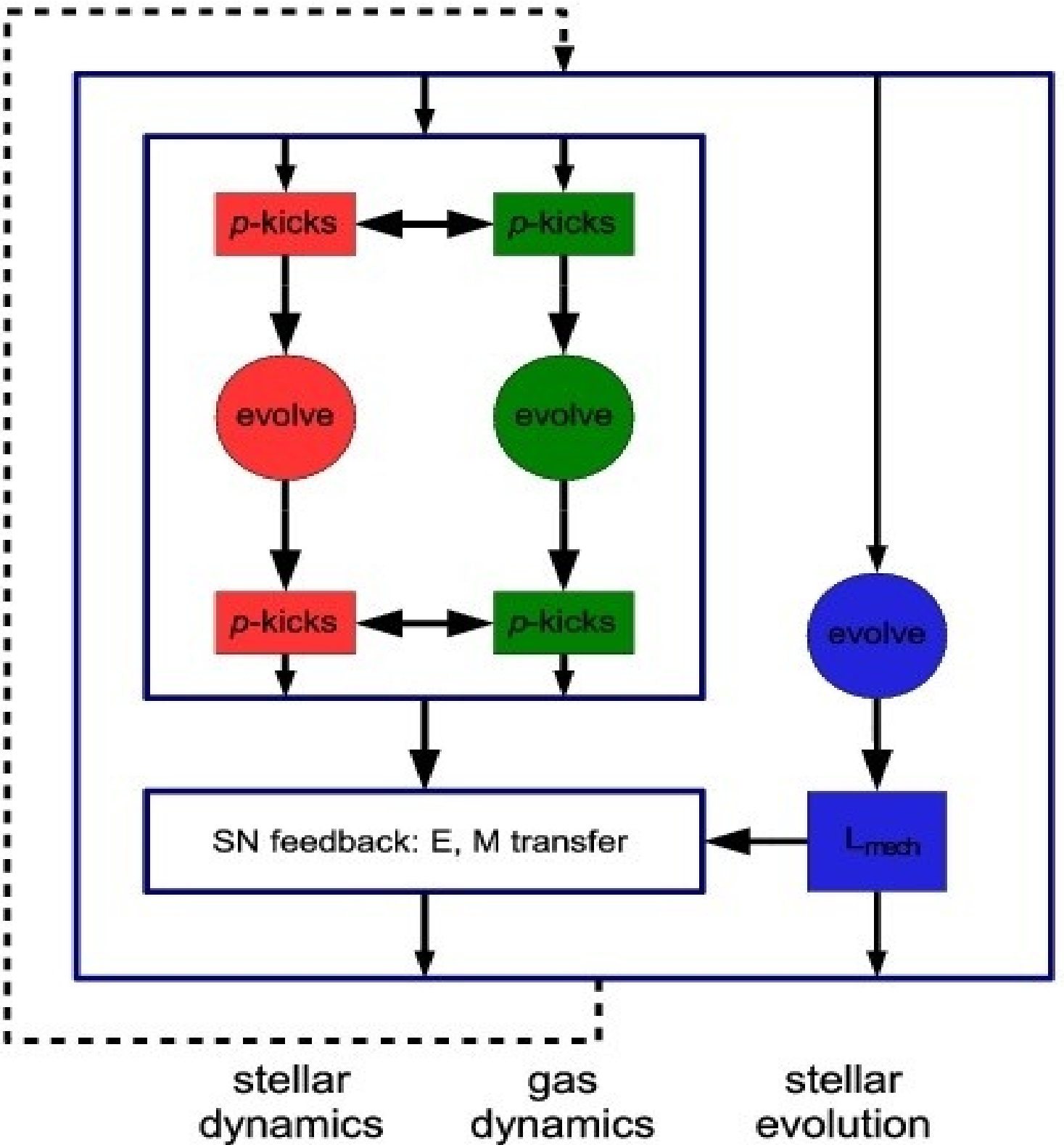
```
> gravity = Gadget2(location=132.229.224.148, nworkers=16)
```



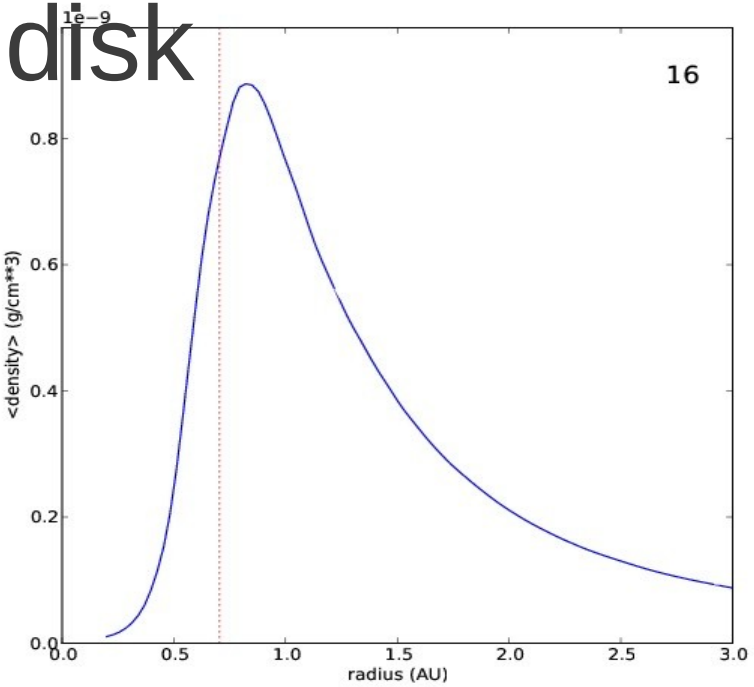

```
> from amuse.lab import *
> bodies = Salpeter(N, Mmin, Mmax)
> stellar = MESA()
> stellar.add_particles(bodies)
> stellar.evolve_model(t_end)
> write_to_file(stellar, “stars.hdf5”)
```

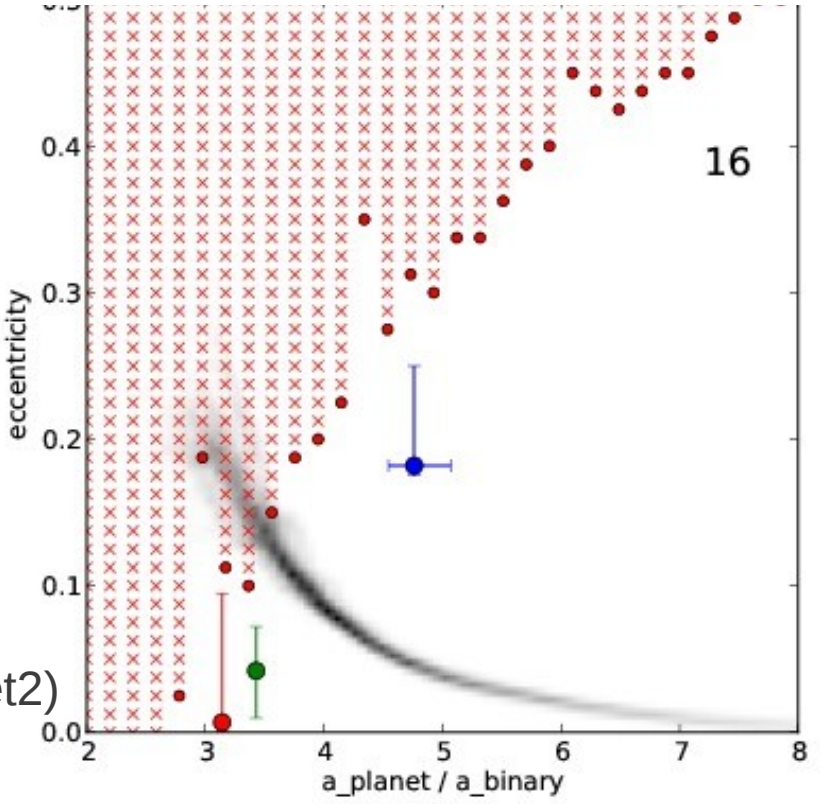


Direct N-body (Hueyno) with stellar evolution (SeBa): by Inti Pelupessy


p-kicks
p-kicks
evolve
evolve
p-kicks
p-kicks
evolve
SN feedback: E, M transfer
L_mech
stellar dynamics
gas dynamics
stellar evolution

Kepler 16 proto-planetary disk





N-body (Var Kepler), stellar ev (MESA) and SPH (Gadget2)

Pelupessy & SPZ MNRAS 2012

Conclusions

- AMUSE is a great tool for studying multi-scale/physics problems in astrophysics
- Applications cover a wide range of topic in astrophysics
- It spawns executable high-performance codes to local or remote machines
- <1% of the wall-clock time is spend in the framework
- But also useful for small experiments, student projects, preliminary studies, outreach...