Finanziato dall'Unione europea NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA
ICSC Centro Nazionale di Ricerca in HPC, Big Data and Quantum Computing

Mercury-Arxes and OPAL: an update

P. Simonetti, D. Polychroni, D. Turrini, R. Politi, S. Ivanovski

! WP1-WP2 monthly meeting !

Spoke 3 II Tec … **op,** Bologna Dec 17 -19, 2024

Parallelizing Mercury-Arxes: scientific rationale

- Observations allow to retrieve the chemical composition of exoplanet atmospheres
- For gas giants, atmospheric composition is a proxy for the composition of the whole body
- From the composition it is possible to reconstruct the formation and evolution of planets
- Investigating these processes in a cosmic context is fundamental to understand these processes

Pipeline of codes to self-consistently model how a planet looks like from its history:

- Chemical composition of the protoplanetary disk
- Planetary formation & migration
- Atmospheric chemistry
- Synthetic spectra

Parameter space exploration is required!

Parallelizing Mercury-Arxes: starting point and objectives

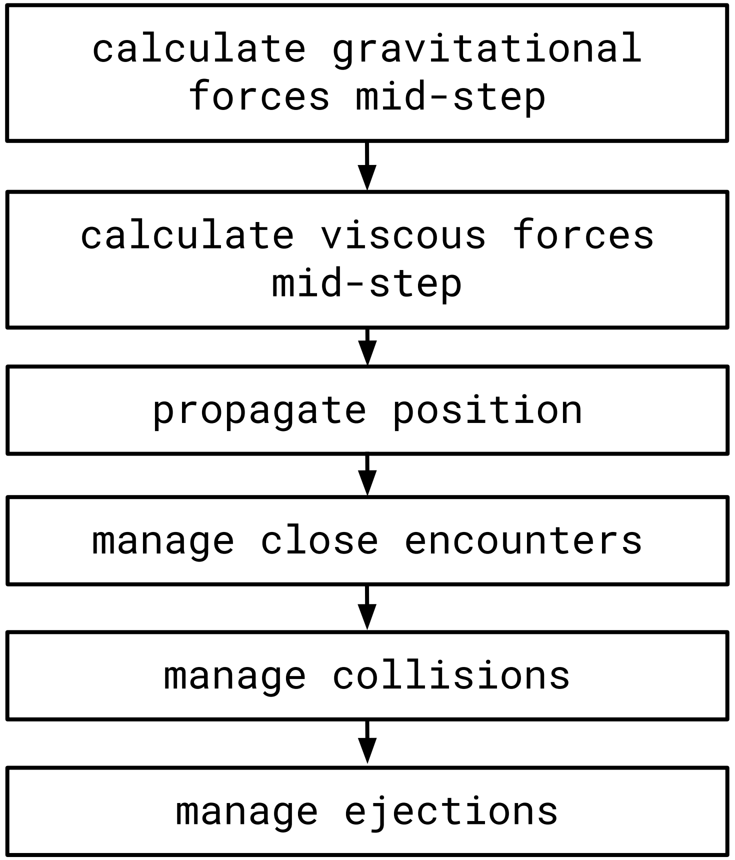
- Mercury (Chambers+99): multipurpose n-body integrator
- Mercury-Arxes (Turrini+21): branched out version tailored to study planetary formation:
  - based on Mercury 6
  - viscous interaction with a time-evolving disk
  - management of merging trees
  - hybrid symplectic integrator to conserve energy and momentum over ~Myr timescales
- Current typical simulation: <10 gravity-source *planets*, >1000 non gravity-source *particles*

GOAL: have all the *particles* as gravity-sources

- especially important at the beginning of the simulation, when most of the mass is in *particles*

Parallelizing Mercury-Arxes: methodology

Mercury-Arxes main loop:

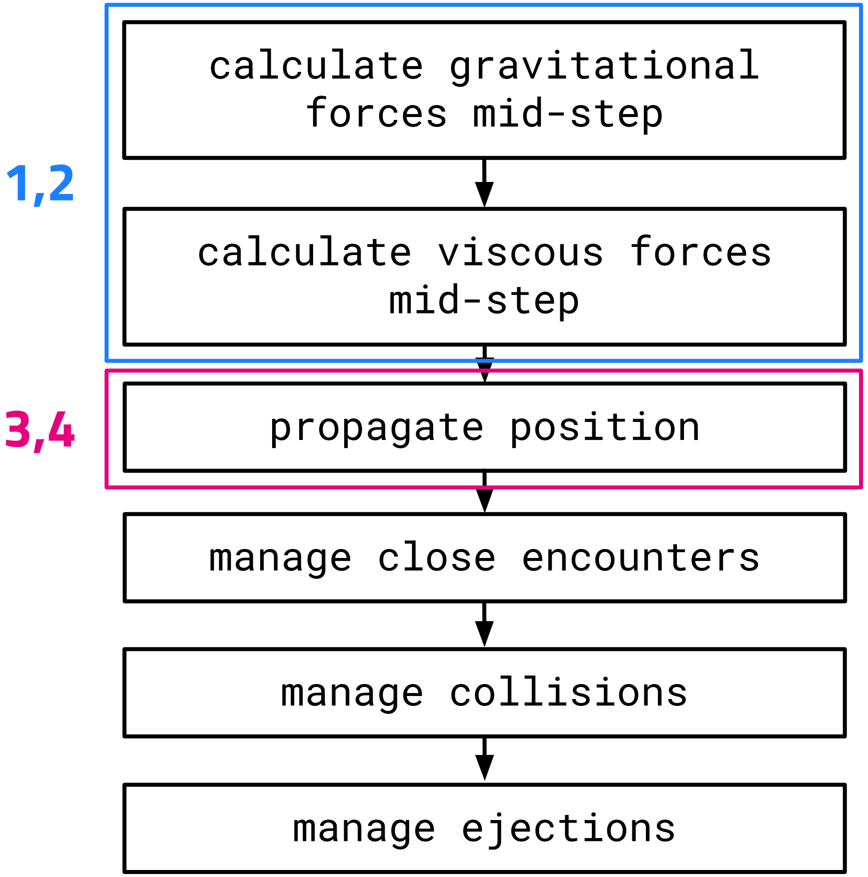


To achieve our goal, high level of parallelism is needed. In practice we:

- ***parallelize the main calculations on GPU using OpenACC***
- ***parallelize the rest of the code on CPU using OpenMP***

Why? Minimal need for code refactoring with respect to the potential gain

Parallelizing Mercury-Arxes: current status

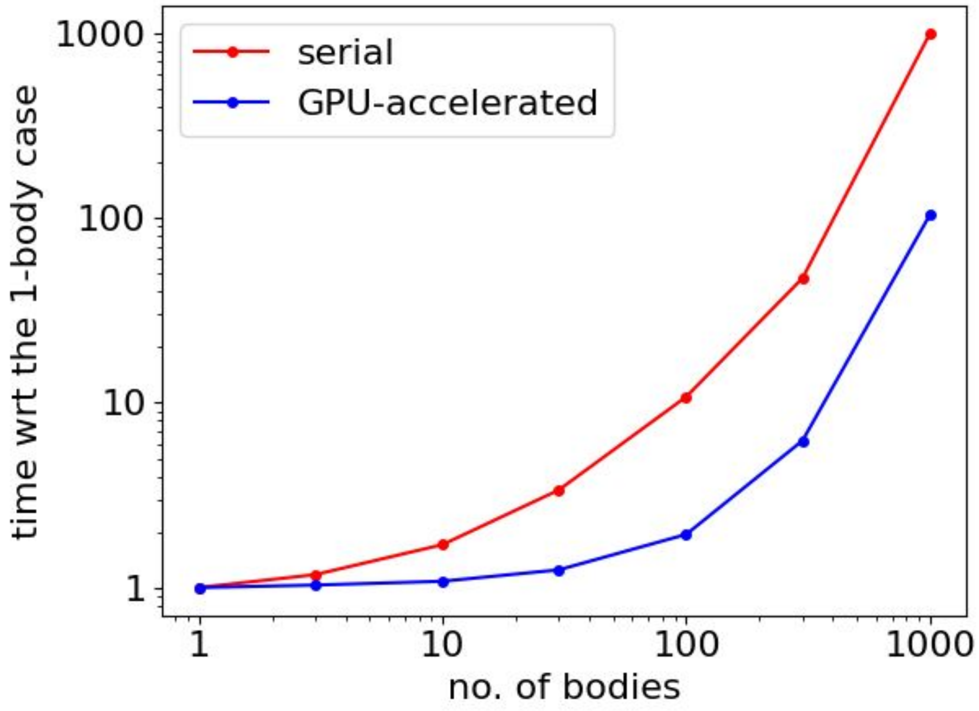


1. **merged gravitational and non-gravitational forces in one routine**
2. **ported force calculation on GPU**
3. **translated the position propagation routines from C to Fortran**
4. **ported position propagation on GPU**

(3) needed because of issues in the interaction between OpenACC and the external C routines - solution suggested by G. Lacopo (Univ. of Trieste, INAF-OATs)

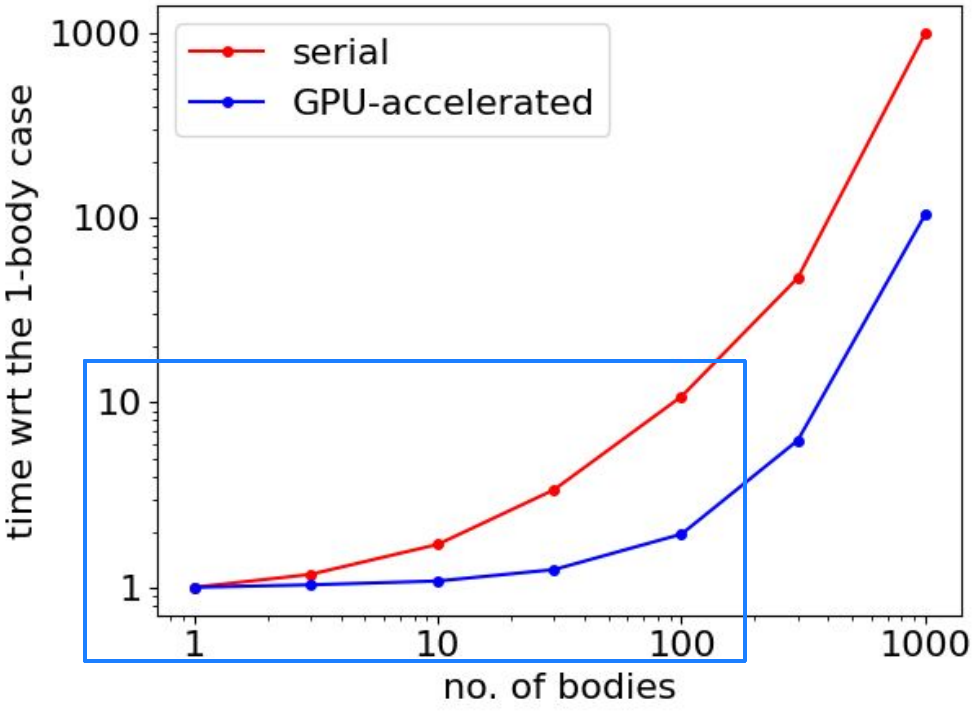
Parallelizing Mercury-Arxes: main results

- marginal speedup from (1) and (3): few % runtime decrease
- large speedup from (2) and (4) despite:
    - inefficient data management
    - inefficient execution of the CPU part due to nvfortran
- substantially improved serviceability of the code



Parallelizing Mercury-Arxes: main results

- marginal speedup from (1) and (3): few % runtime decrease
- large speedup from (2) and (4) despite:
    - inefficient data management
    - inefficient execution of the CPU part due to nvfortran
- substantially improved serviceability of the code



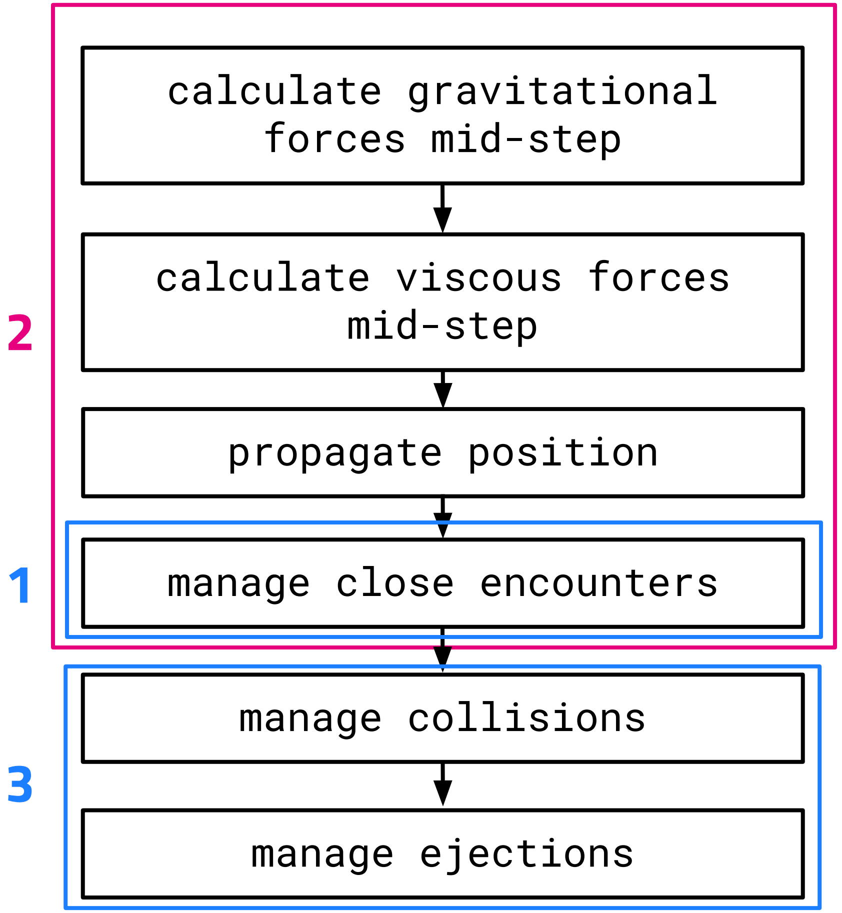
Fewer bodies → fewer (non-parallel) close encounters → larger advantage of the GPU-accelerated version

Parallelizing Mercury-Arxes: KPI & milestones

- **Small delay with respect to the M9 target related to Mercury-Arxes of OPAL, mostly because of the unforeseen need for code refactoring**
- **Assessed progression on this target: 75%**

ID	TARGET	KPI	MILESTONE
4	Assessment of the comparative performances of the OpenMP and OpenACC releases of Mercury-Arxes (**Task f**)	Report on the HPC assessment on Mercury-Arxes	M9

Mercury-Arxes: final steps



1. **try to parallelize close encounter management - trickier than the rest of the loop!**
2. **define a data region in order to cut down transfer times btw. host and device**
3. **parallelize collision and ejection management via OpenMP (minor)**

ENDPOINT: comparative performance assessment and deployment of the fully parallelized version of the code

What is OPAL:

A **PNRR Key Science Project** with two goals:

- produce a library of detailed synthetic atmospheric models of giant planets for the Ariel space mission (Tinetti+2018, 2020; Turrini+2018, 2022)
- port the Arχes planet formation suite of codes to HPC

Takes advantage of allocated time in:

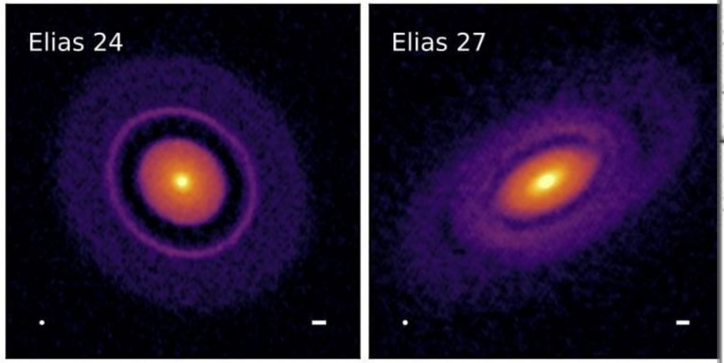
Leonardo (4 million core hours)
Pleiadi (2.5 million core hours)

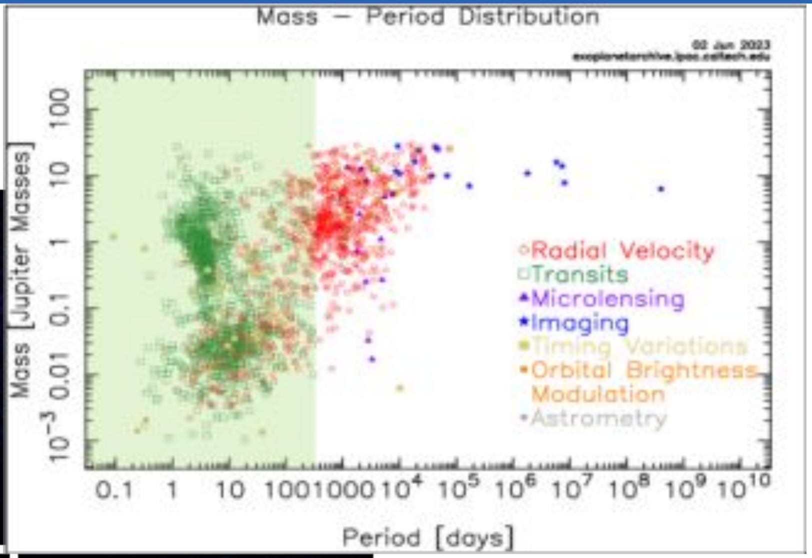
as well as the dedicated cluster **GENESIS+**.

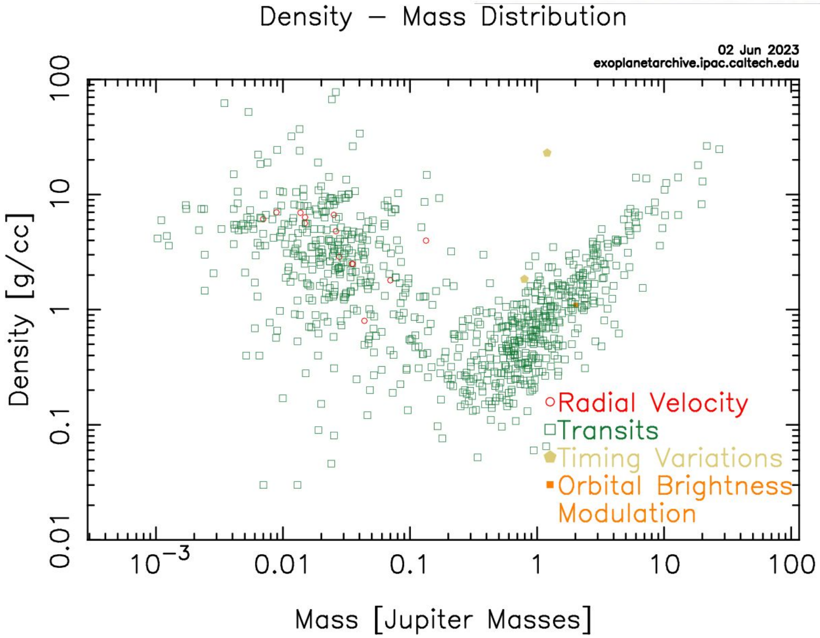
OPAL is a coordinated effort between the ARIEL, INAF and the PNRR

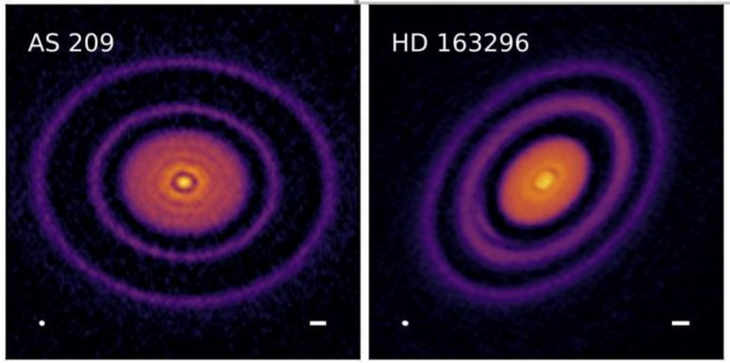
Why we need it:

- New observational data highlight our limited understanding of where planets form in disks





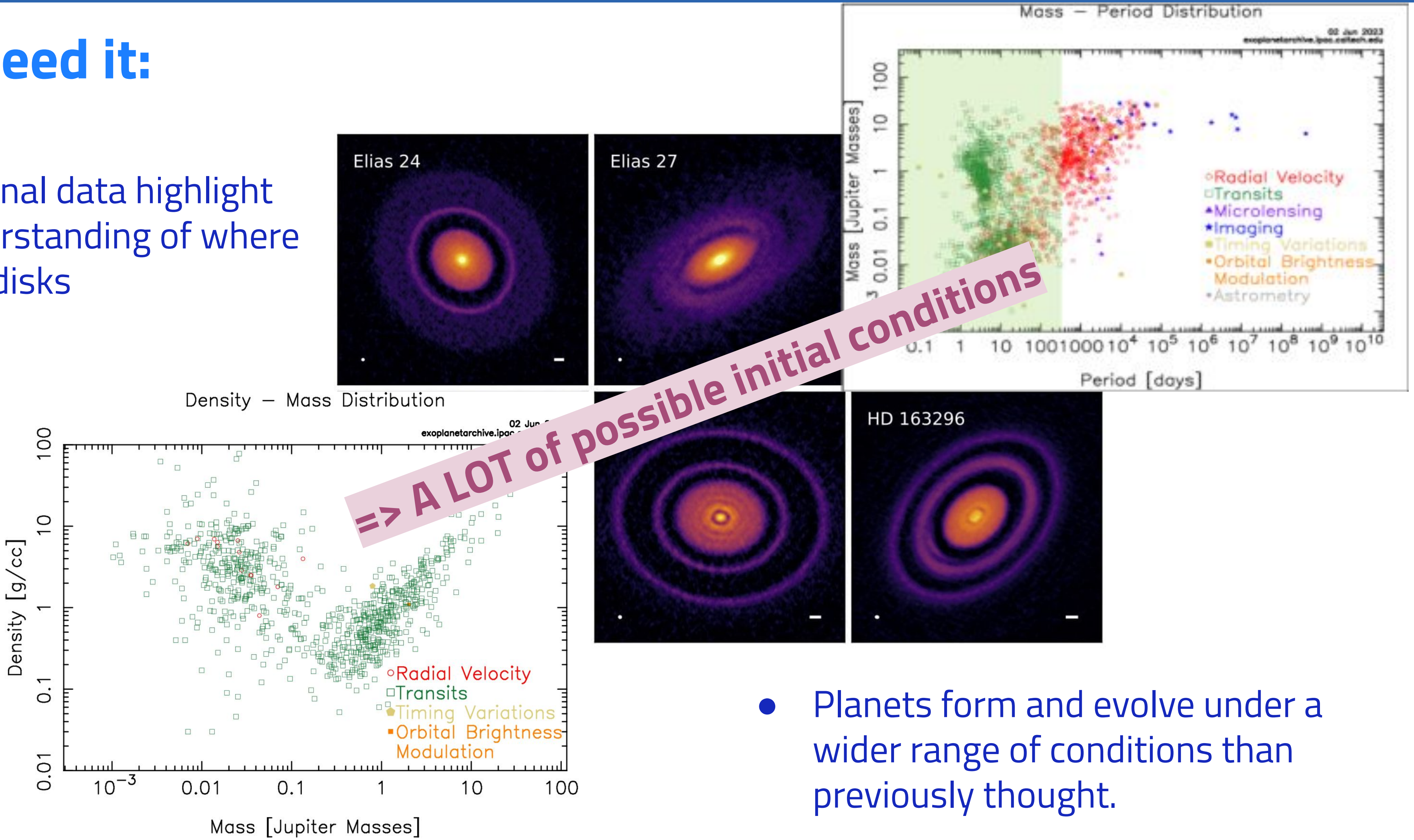




- Planets form and evolve under a wider range of conditions than previously thought.

Why we need it:

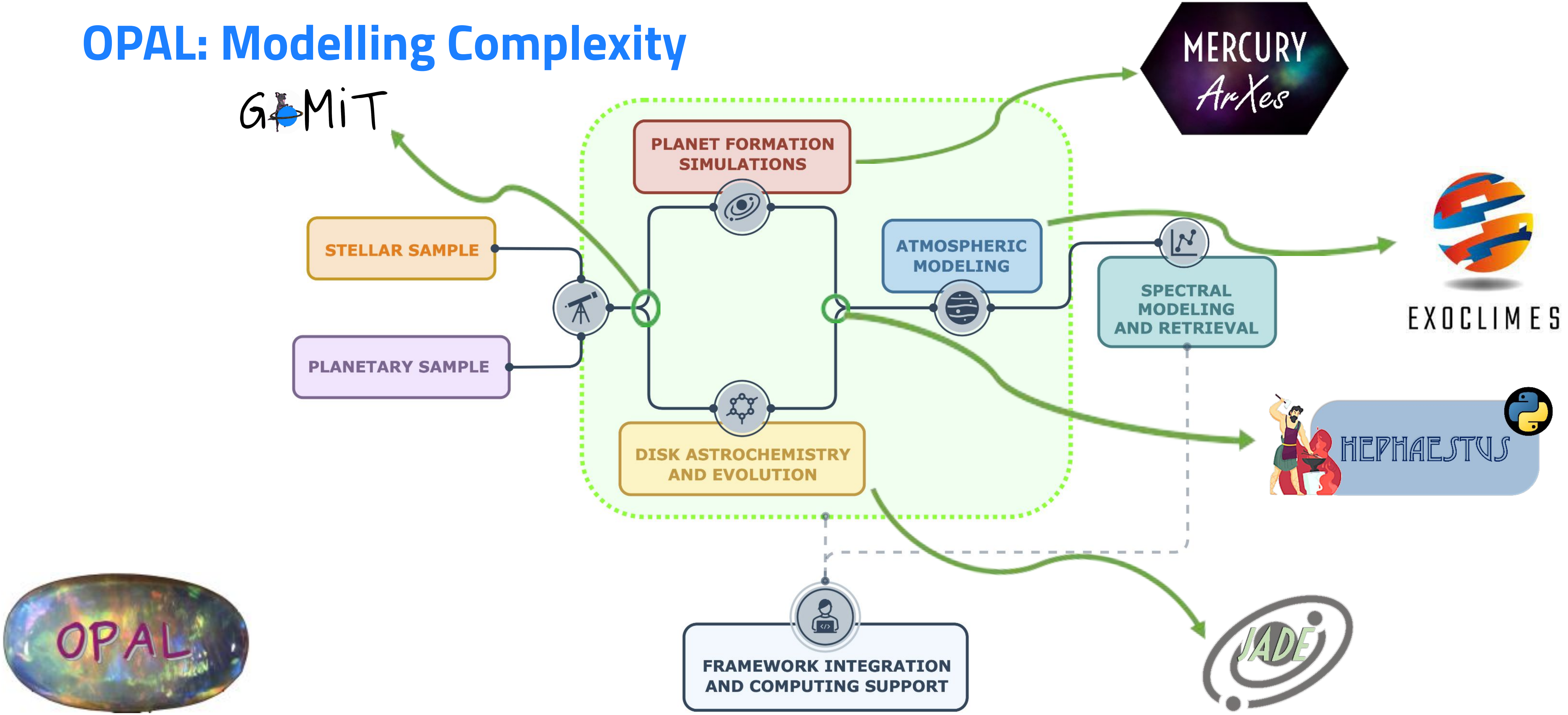
- New observational data highlight our limited understanding of where planets form in disks



=> A LOT of possible initial conditions

- Planets form and evolve under a wider range of conditions than previously thought.

OPAL: Modelling Complexity



OPAL: Modelling Complexity

JADE (Pacetti et al. in prep.) is the multi-language (Python + Fortran) data-parallel code to jointly simulate the **physical and chemical evolution of protoplanetary disks** of the *Arχes* suite.

Mercury-Arχes (Turrini+2019,2021) is the parallel n-body code of the Arχes suite incorporating physical libraries to simulate **planet formation in protoplanetary disks**

HEPHAESTUS (Turrini+2021; Pacetti+2022) is the Python compositional post-processing tool of the *Arχes* suite that resolves planetary compositions across 20+ elements.

FastChem and **Vulcan** are atmospheric modelling codes from the University of Bern's ExoClimes suite convert Hephaestus' elemental compositions into atmospheric molecular compositions (e.g. Fonte+2023, Simonetti et al., under review).

OPAL: Modelling Complexity

e to jointly
uite.

Mercur
physical

processing
nents.

An extra word on Vulcan:

We are expanding the chemical network of Vulcan to also include the sedimentation and transport of the condensation of iron!

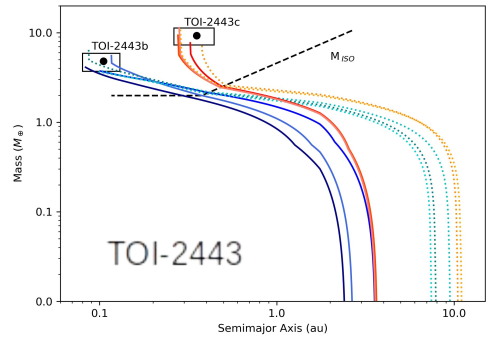
(and soon also the silicates!)

FastChem and **Vulcan** are atmospheric modelling codes from the University of Bern's ExoClimes suite convert Hephaestus' elemental compositions into atmospheric molecular compositions (e.g. Fonte+2023, Simonetti et al., under review).

GroMiT: The new kid in town!

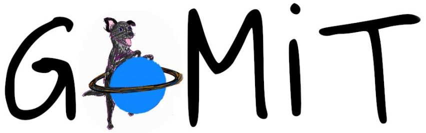
GroMiT, the Growth and Migration Track code, tracks how a forming planet accretes pebbles and gas as well as how it migrates depending on the local environment of its native circumstellar disk (e.g. Johansen+2019, Ida+2016, Tanaka+2020).

Born as a plug-in for Mercury-Arχes to make it self-consistent it found plenty of applications also as a stand-alone code in exploring the formation histories of **observed planets (GAPS)**.



It comes in two flavours!

→ Single Planet Mode (Polychroni+2023)



→ Population Synthesis Mode (Polychroni+ in prep)