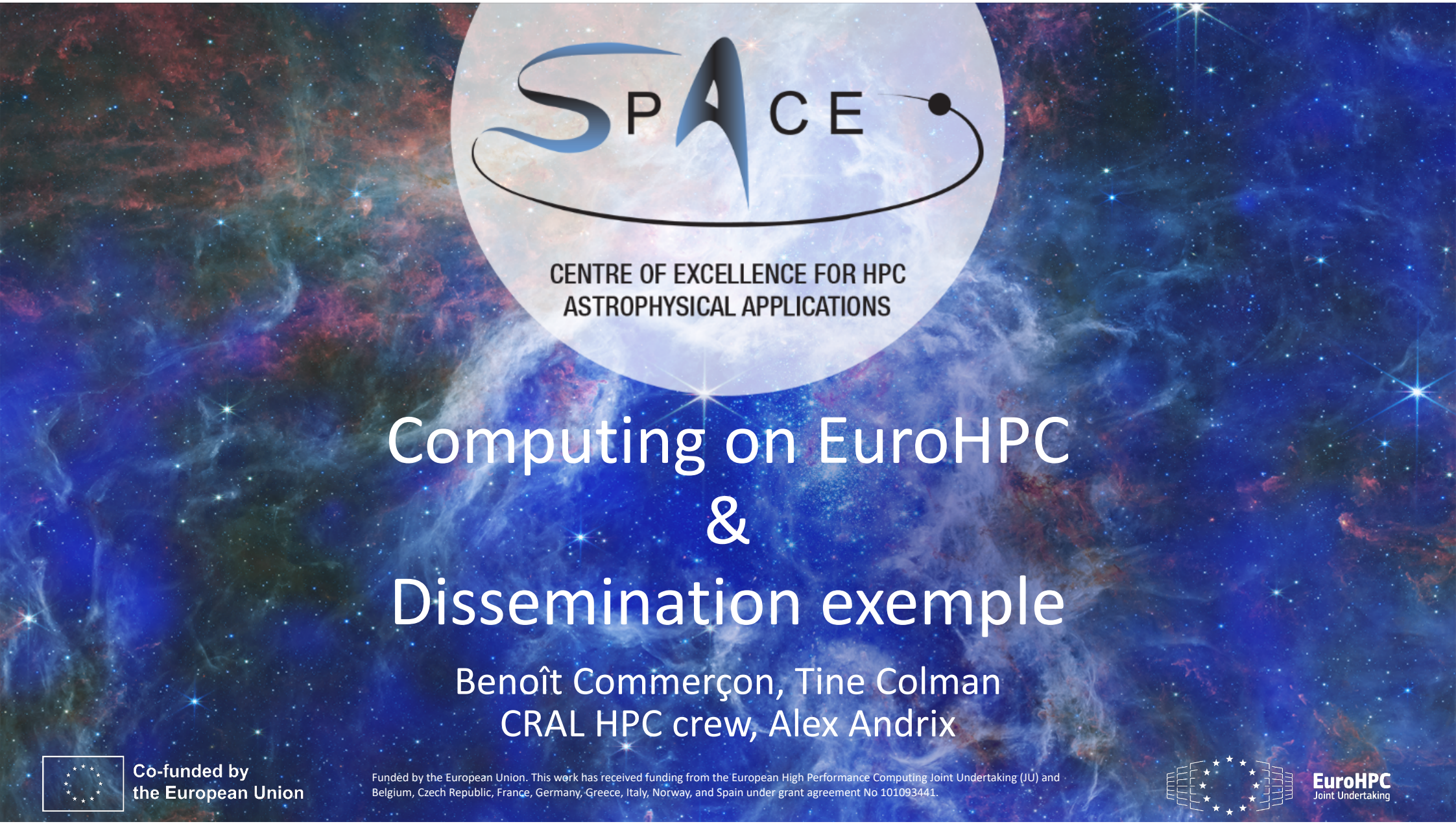


Computing on EuroHPC
&
Dissemination exemple

Benoît Commerçon, Tine Colman
CRAL HPC crew, Alex Andrix

Co-funded by the European Union

Funded by the European Union. This work has received funding from the European High Performance Computing Joint Undertaking (JU) and Belgium, Czech Republic, France, Germany, Greece, Italy, Norway, and Spain under grant agreement No 101093441.

EuroHPC Joint Undertaking

EuroHPC calls

Development - develop, test & optimise

Cut-off once a month

4 000 Node hours (~500 000 hCPU)

Benchmark - test & benchmark

Cut-off once a month

2 000 Node hours (~250 000 hCPU)

Regular

2 per year (March and September)

Minimum 60 000 Node hours (~8M hCPU)

Extreme

2 per year (April and October)

Minimum 130 000 to 240 000 Node hours (~16M hCPU)

https://www.eurohpc-ju.europa.eu
https://hpc-portal.eu/

CENTRE DE RECHERCHE ASTROPHYSIQUE DE LYON

Deployment on JU systems

JU systems

	LUMI	LEONARDO	MNOSTR V	MELUXINA	KAROLINA	VEGA	DEUCALION	DISCOVER ER
Perf	386 PFlops	249 PFlops	215 PFlops	12.8 PFlops	9.6 PFlops	7 PFlops	7.5 PFlops	4.5 PFlops
CPU	AMD EPYC (2048 nodes)	Intel Saphire Rapids (1536 nodes)	Intel Saphire Rapids (6400 nodes)	AMD EPYC (570 nodes)	AMD EPYC (800 nodes)	AMD EPYC (1020 nodes)	A64FX (1600 nodes), AMD EPYC (500 nodes)	AMD EPYC (1100 nodes)
GPU	4 AMD MI250X (300 nodes)	4 A100 (3456 nodes)	4 Hopper (1120 nodes)	4 A100 (200 nodes)	8 A100 (72 nodes)	4 A100 (60 nodes)	4 A100 (33 nodes)	4 H200 (4 nodes)

CENTRE DE RECHERCHE ASTROPHYSIQUE DE LYON

Deployment on JU systems

Deployed (compile&run) on *8* out of 8 JU systems

	LUMI		LEONARDO		MNOSTR V		MELUXINA		KAROLINA		VEGA		DEUCALION		DISCOVER ER
access	✓		✓		✓		✓		✓		✓		✓		✓
	cpu	gpu	cpu	gpu	cpu	gpu	cpu	gpu	cpu	gpu	cpu	gpu	cpu	gpu	cpu
ported	✓	✓	✓	-	✓	-	✓	-	✓	-	✓	-	✓	-	✓
benchmarked	✓	✓	✓	-	✓	-	✓	-	✓	-	✓	-	-	-	✓

CPU: RAMSES

GPU: SHAMROCK

WG Benchmark
WG Scripts Database

CENTRE DE RECHERCHE ASTROPHYSIQUE DE LYON

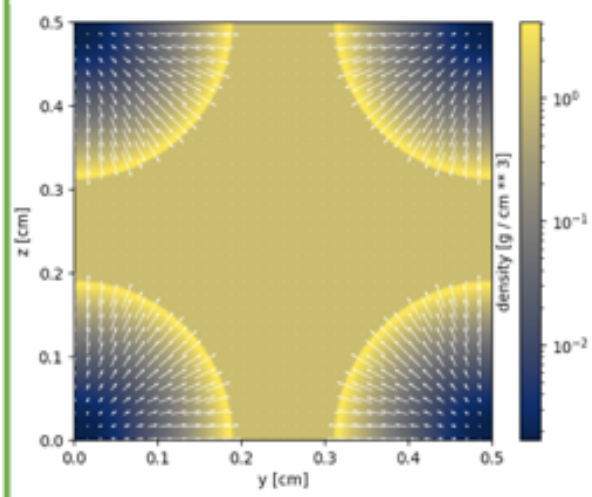
EuroHPC systems (CPU partitions)

system (country)	#nodes	RAM/node [GB]	processor (cores)	chosen compiler
Discoverer (Bulgaria)	1128	256, 1024	2 x AMD EPYC 7H12 (2x64)	GNU + OpenMPI
Karolina (Czech rep.)	720	256	2 x AMD EPYC 7H12 (2x64)	GNU + OpenMPI
Vega (Slovenia)	960	256, 1024	2 x AMD EPYC 7H12 (2x64)	GNU + OpenMPI
MeluXina (Luxembourg)	573	512, 4096	2 x AMD EPYC 7H12 (2x64)	GNU + OpenMPI
Lumi (Finland)	2048	256, 512, 1024	2 x AMD EPYC 7763 (2x64)	GNU + MPICH (Cray)
Leonardo (Italy)	1536	512	2 x Intel Xeon Platinum 8480+ (2x56)	Intel + Intel MPI
Mare Nostrum (Spain)	6192	256, 1024	2 x Intel Xeon Platinum 8480+ (2x56)	Intel + Intel MPI
Deucalion (Portugal)	-	-	-	-

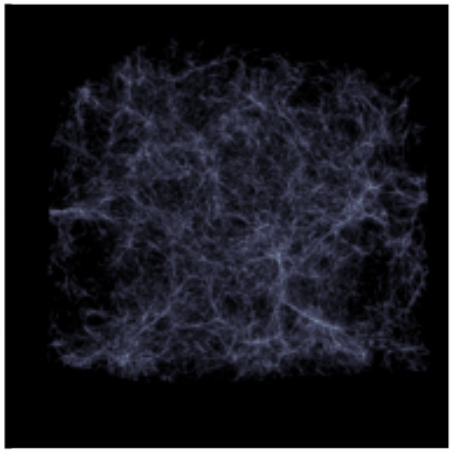
SPACE benchmarks

Selected representative use cases:

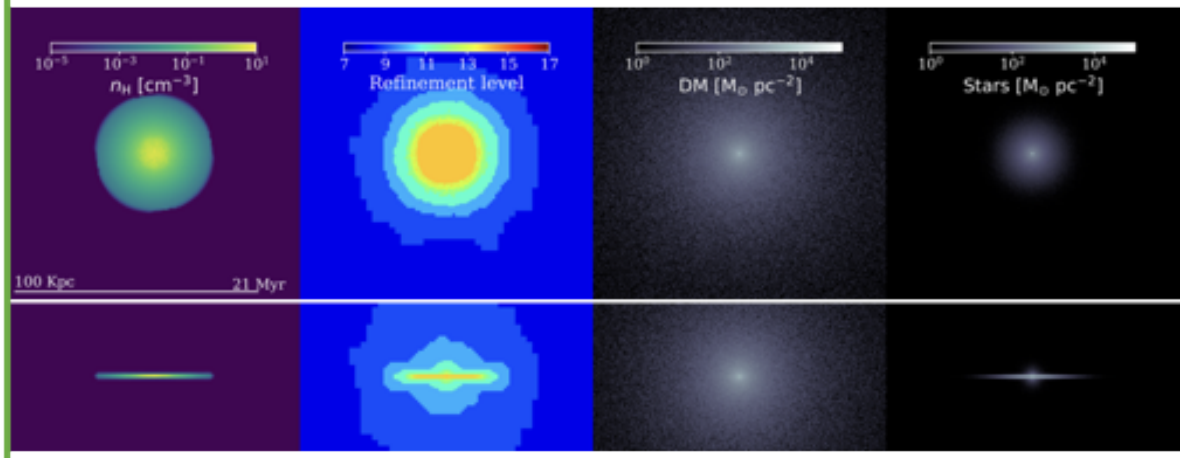
Sedov explosion



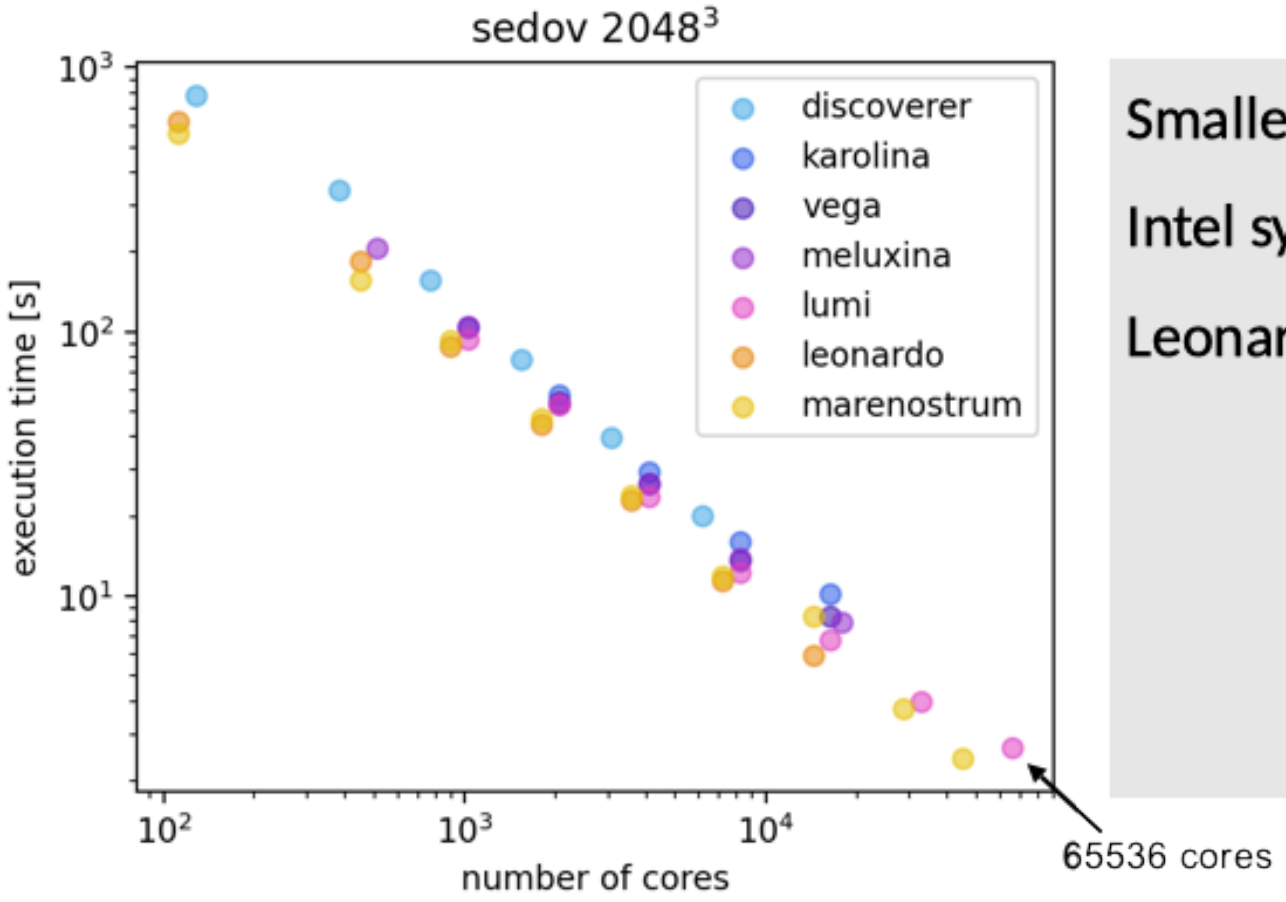
Cosmological box



Isolated galaxy



Performance on EuroHPC systems

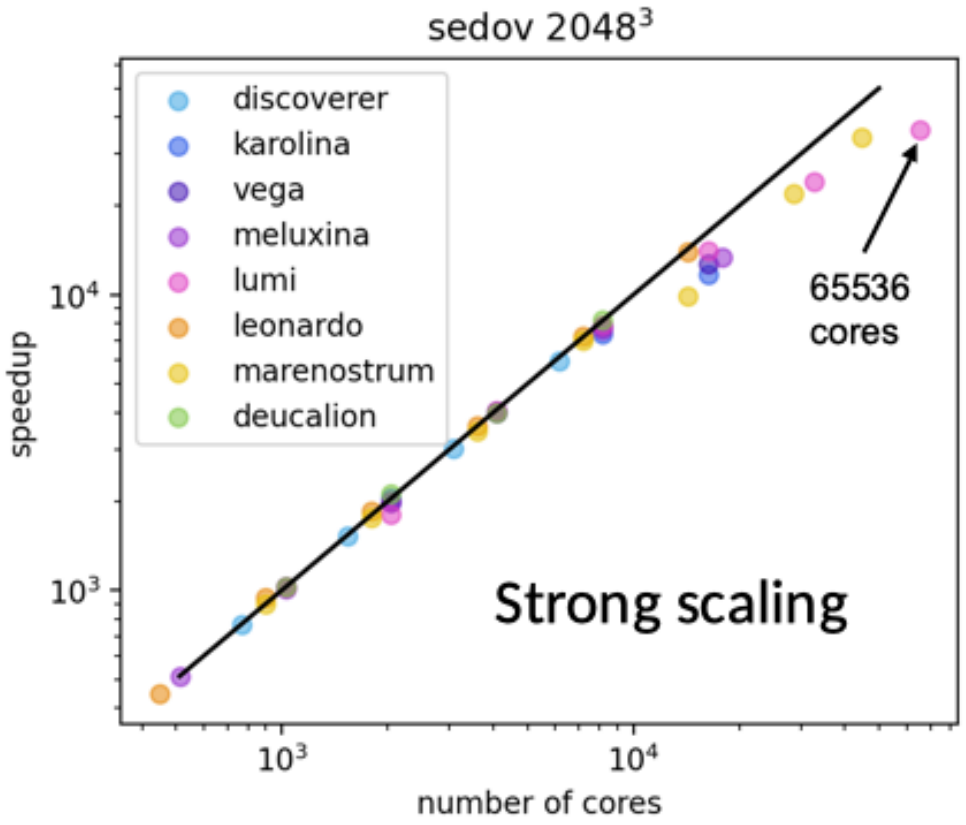


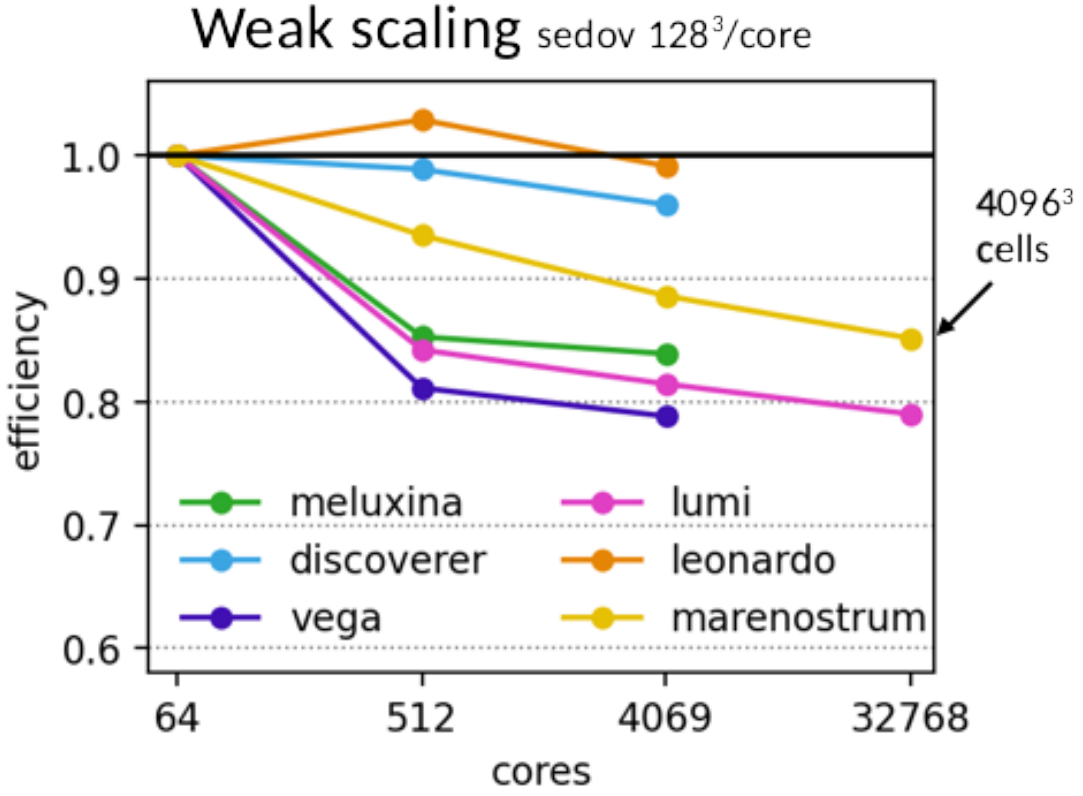
Smaller AMD systems similar

Intel systems typically slightly faster

Leonardo best timings (but unstable)

Scaling on EuroHPC systems





Performance benchmarking

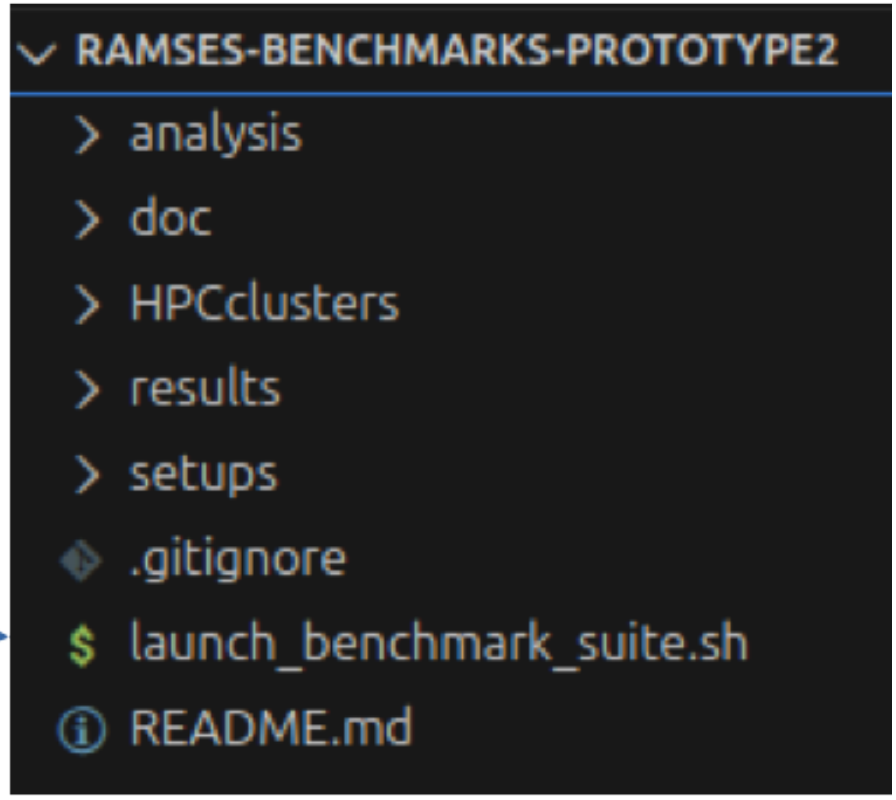
GOAL: create script to make benchmarking ramses easy...

- for users, that need scaling plots for proposals
- for developers, that need to verify performance
- for maintainers, to verify code quality on PR

Structure of the benchmark directory

https://github.com/tinecolman/ramses-benchmarks-prototype2

- Scripts to analyze results
- Manual
- Presets for different clusters
- Results and figures
- Collection of predefined setups
- **Main script**: compile code, create and launch jobs
- Front page of repo



Using the benchmark script

```
$ ./launch_benchmark_suite.sh -c meluxina -l "1" -h refactor_3cube_nbor_utils -t 1,2 -a p200525 -i 3
```

Parameters:

- -c: name of the preset cluster
- -l or -n: specify a list or maximum number of nodes to use
- -t: specify which benchmark setups to run
- -i: how many times to run the same benchmark (1 by default)
- -a: allocation number (can also be determined automatically)
- -w: also do weak scaling
- -m: openMP threads

```
Switched to branch 'openmp_hydro_unigrid'
Your branch is up to date with 'origin/openmp_hydro_unigrid'.
################################################
#    Launching RAMSES performance benchmarks   #
################################################
Multiple projects found. Please select one:
1) nocredit
2) p200525
#? 2
You selected allocation ID: p200525
Will launch the following benchmarks:
 [ 2] setups/sedov
----------------------------------------
Test 1/1: sedov
Compile job submitted with Job ID: 2577666
Waiting for compile job to finish...
Compile job completed successfully.
Launched benchmark sedov on 1 nodes [JOB ID 2577669]
Launched benchmark sedov on 1 nodes [JOB ID 2577670]
Launched benchmark sedov on 1 nodes [JOB ID 2577671]
Launched benchmark sedov on 2 nodes [JOB ID 2577672]
Launched benchmark sedov on 2 nodes [JOB ID 2577673]
Launched benchmark sedov on 2 nodes [JOB ID 2577674]
Launched benchmark sedov on 4 nodes [JOB ID 2577675]
Launched benchmark sedov on 4 nodes [JOB ID 2577676]
Launched benchmark sedov on 4 nodes [JOB ID 2577677]
Submitted batch job 2577678
```

Resulting output on scratch

```
[u102032@login03 myruns]$ ls
ICs
benchmark_HEAD_2025-03-04_7c2b0363
benchmark_HEAD_2025-03-04_9e7b310b
benchmark_HEAD_2025-03-04_cce4cf97
benchmark_openmp_2025-03-17_3586df4a
benchmark_oper
benchmark_op
benchmark_oper
benchmark_oper
benchmark_per
```

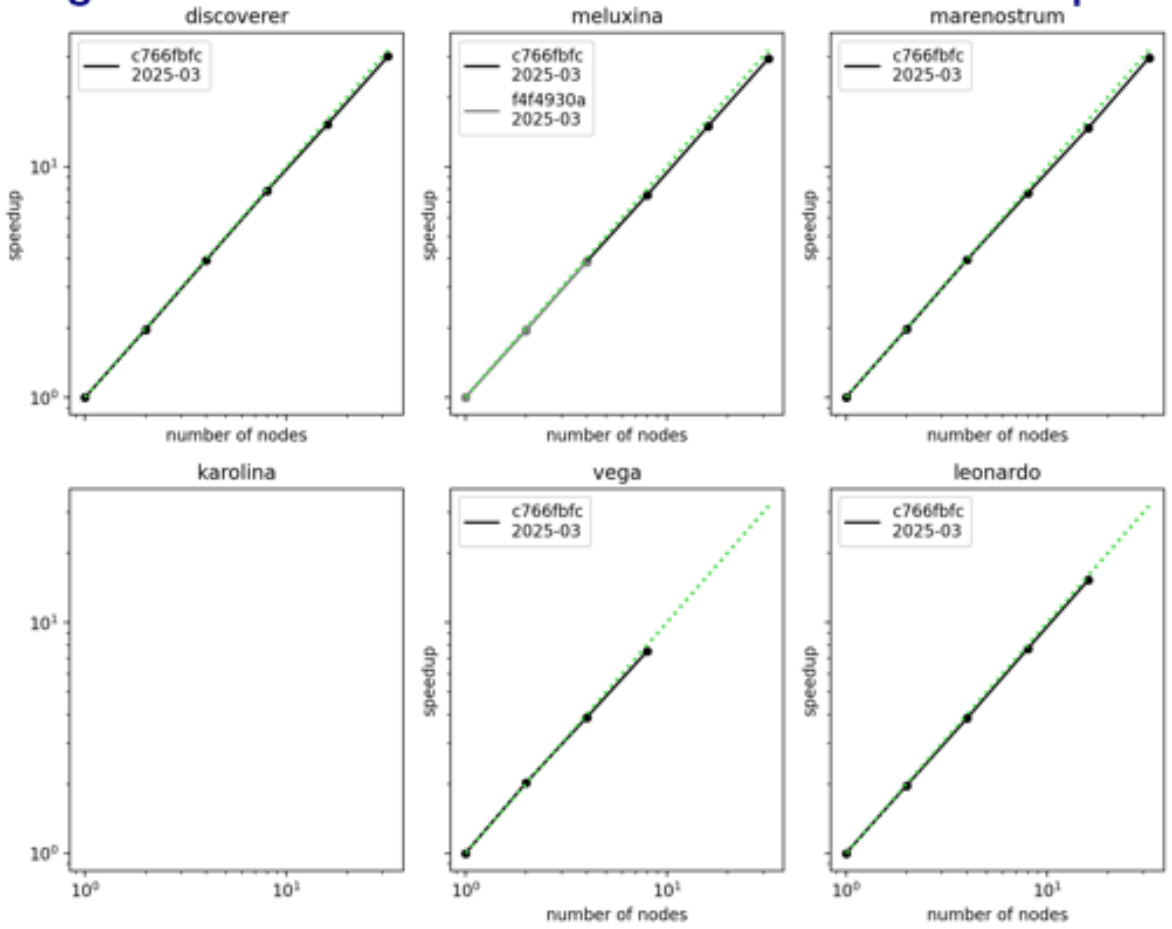
```
vd
/benchmark_openmp_2025-03-19_c766fbfc/sedov

525 4096 Mar 19 12:15 nodes1_reso1024_omp0
525 4096 Mar 19 12:15 nodes1_reso1024_omp1
525 4096 Mar 19 12:15 nodes1_reso256_omp0
525 4096 Mar 19 12:15 nodes1_reso256_omp1
525 4096 Mar 19 12:15 nodes2_reso1024_omp0
525 4096 Mar 19 12:15 nodes2_reso1024_omp1
525 4096 Mar 19 12:15 nodes4_reso1024_omp0
525 4096 Mar 19 12:15 nodes4_reso1024_omp1
525 4096 Mar 19 12:15 nodes8_reso512_omp0
525 4096 Mar 19 12:15 nodes8_reso512_omp1
```

Now get timings from logs…

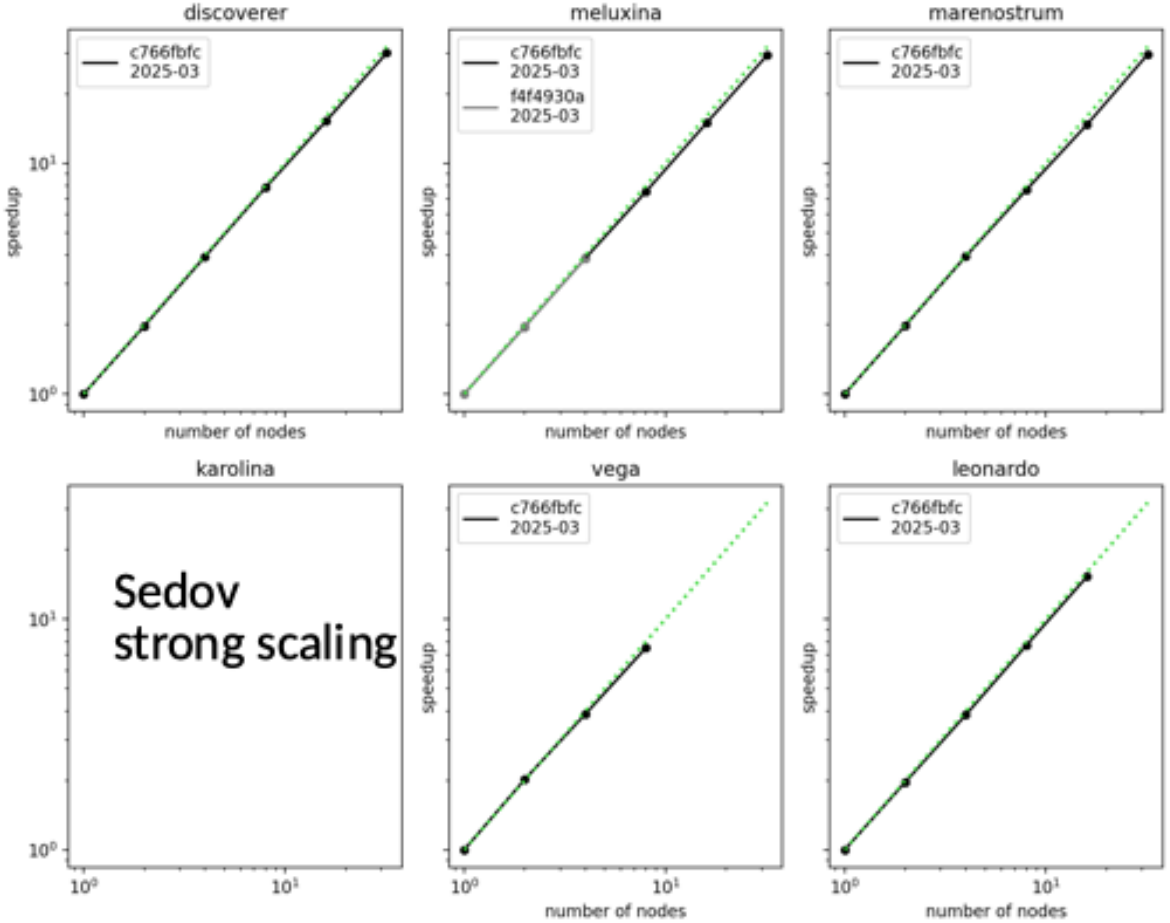
```
Mar 19 12:15 benchmark_exe_3d
Mar 19 12:15 job.sh
Mar 19 12:17 run_2025-03-19_12h15_2571902.log
Mar 19 12:17 run_2025-03-19_12h15_2571903.log
Mar 19 12:17 run_2025-03-19_12h15_2571904.log
Mar 19 12:15 sedov_1024.nml
Mar 19 12:17 slurm_2571902.err
Mar 19 12:15 slurm_2571902.out
Mar 19 12:17 slurm_2571903.err
Mar 19 12:15 slurm_2571903.out
Mar 19 12:17 slurm_2571904.err
Mar 19 12:15 slurm_2571904.out
```

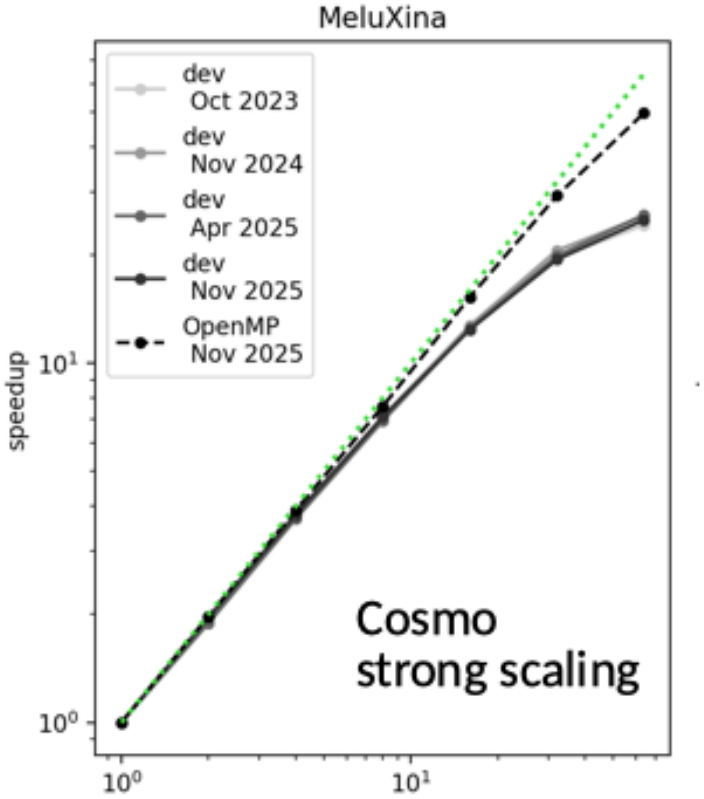
Examples: strong scaling of sedov

https://github.com/tinecolman/ramses-benchmarks-prototype

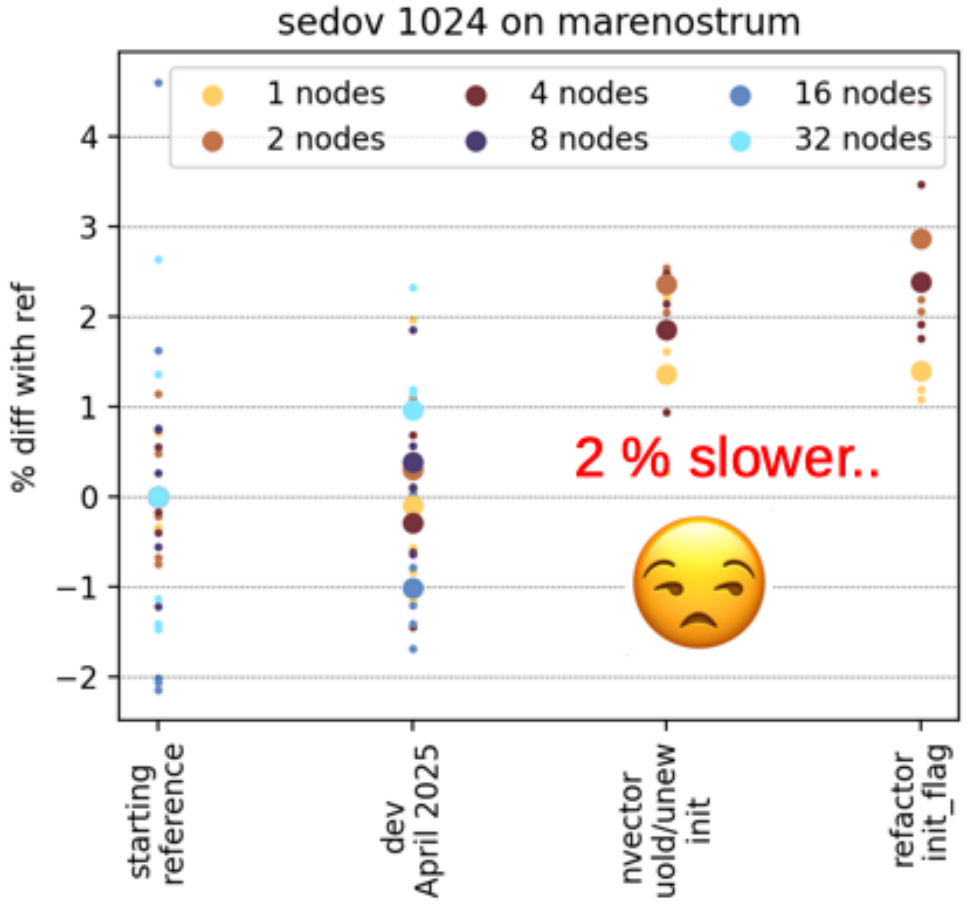


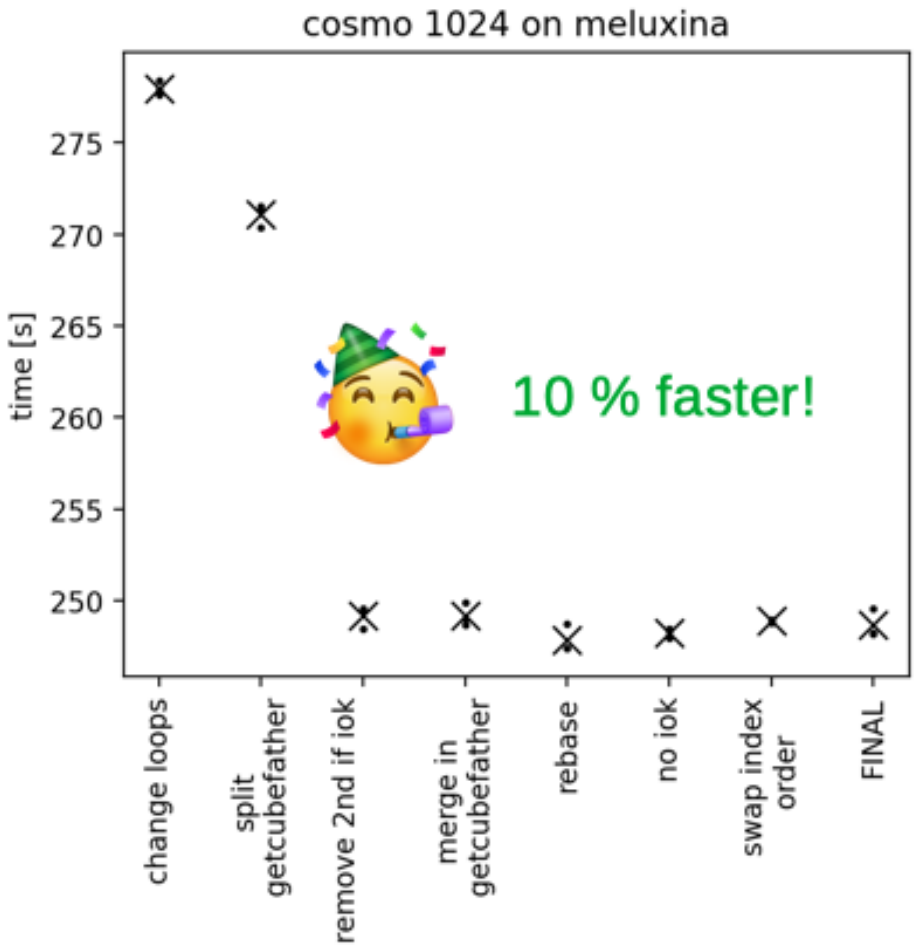
Examples: scaling



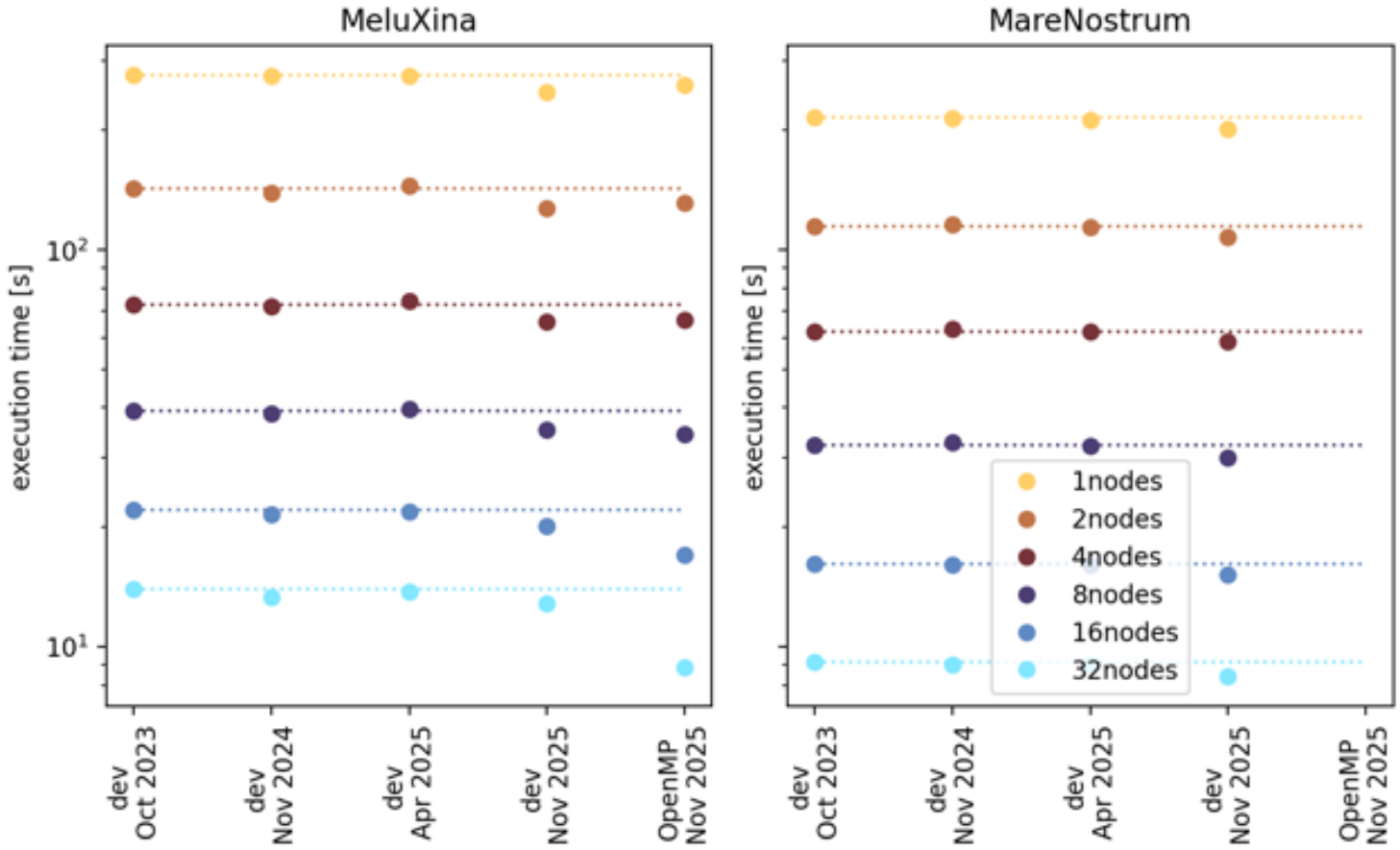


Examples: testing optimization

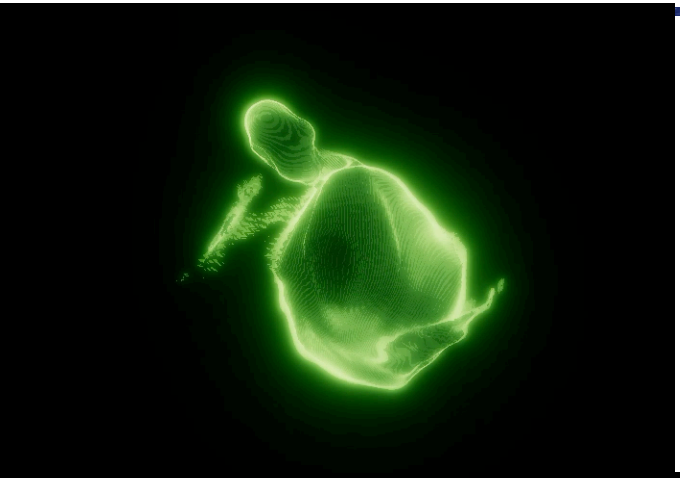




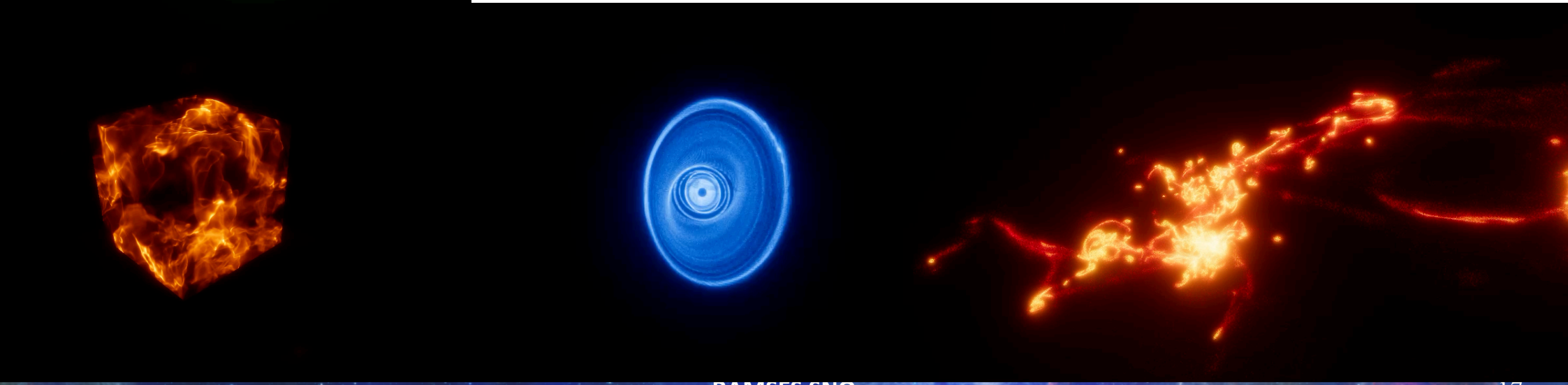
Examples: monitoring performance



Dissemination

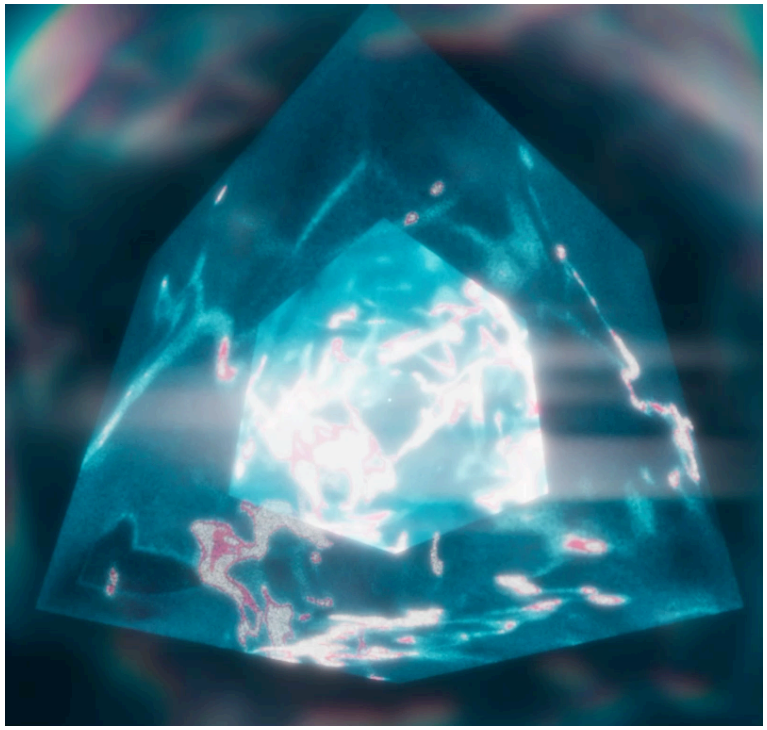


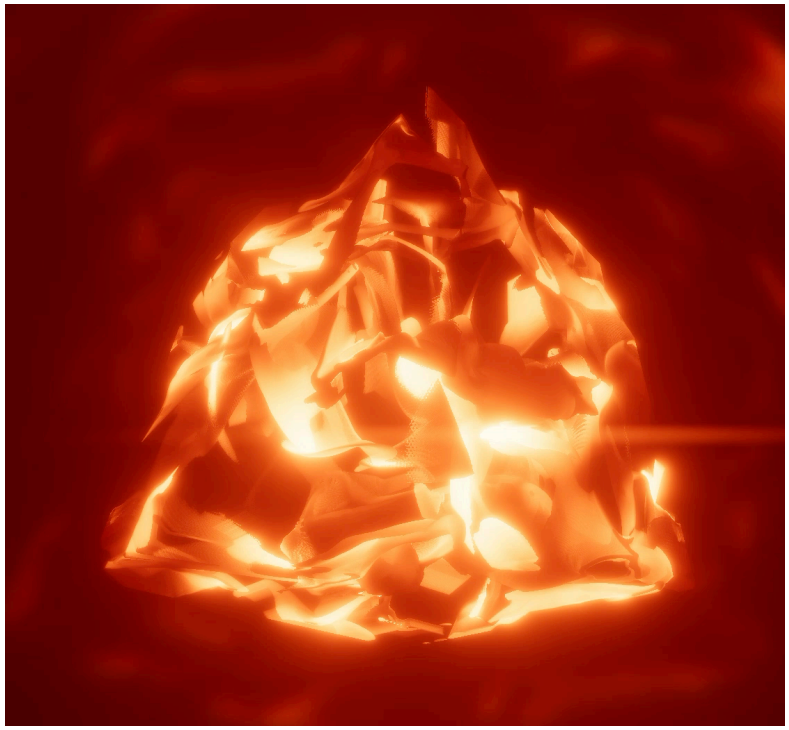
Collaboration with Alex Andrix
https://alexandrix.com/

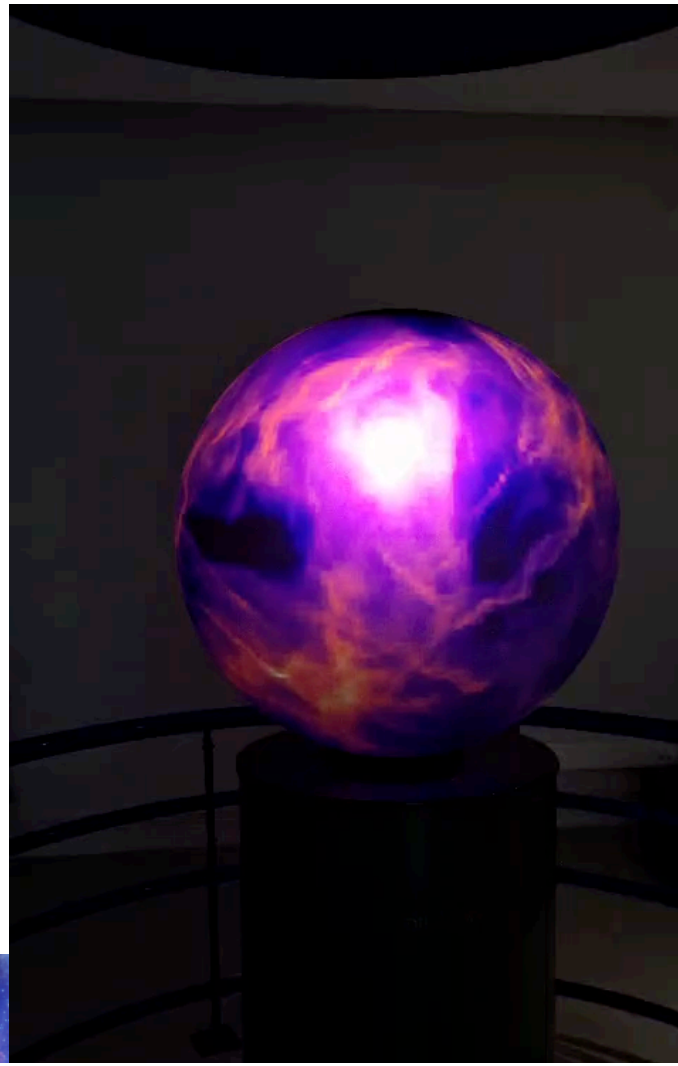


Dissemination

Collaboration with Alex Andrix
https://alexandrix.com/







Acknowledgement & Disclaimer

Funded by the European Union. This work has received funding from the European High Performance Computing Joint Undertaking (JU) and Belgium, Czech Republic, France, Germany, Greece, Italy, Norway, and Spain under grant agreement No 101093441.