

DDF Benchmarking at LAB

SE2I Projects:

- **Exploring the solar system**

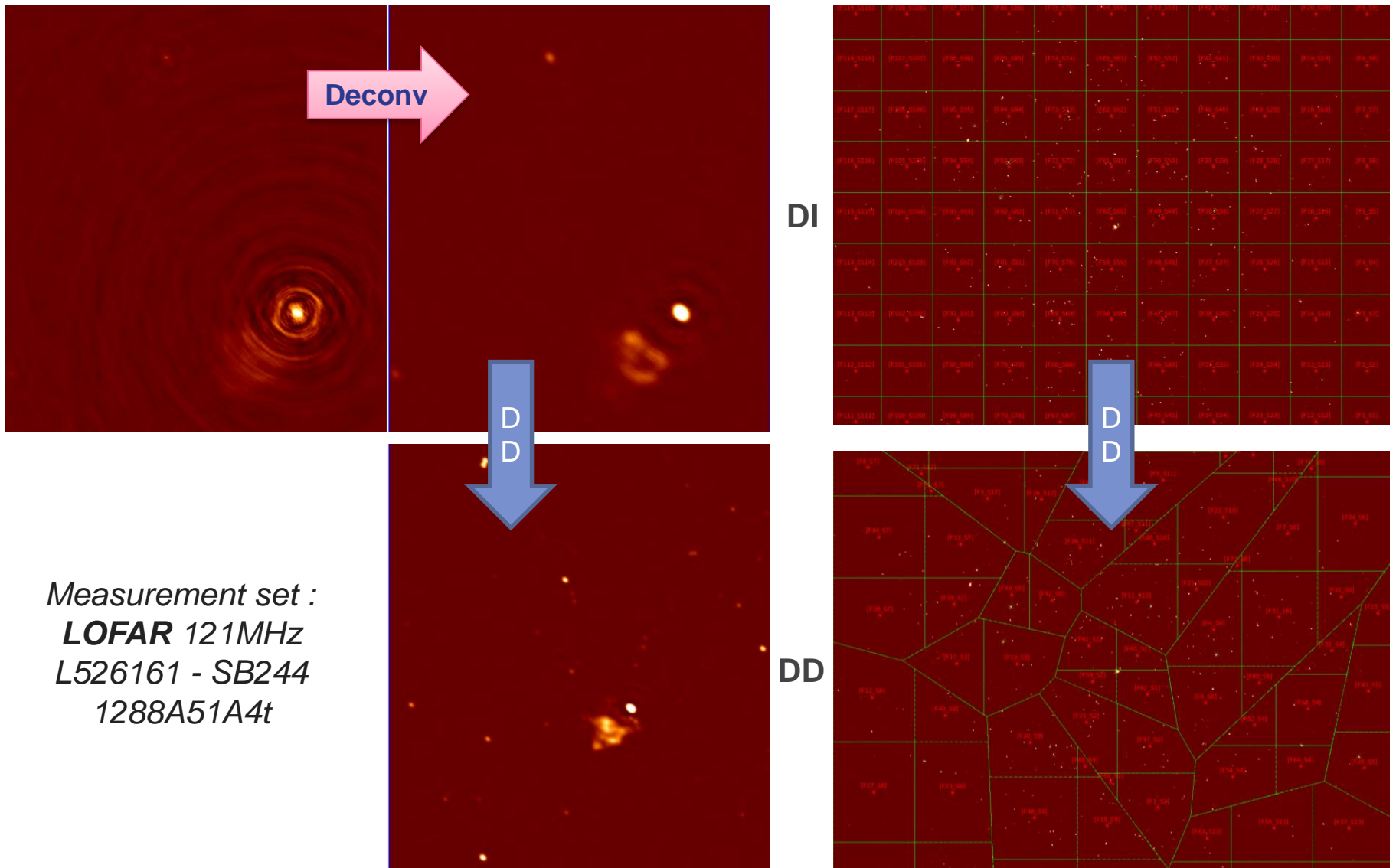
ChemCam (Curiosity), SuperCam (Perseverance), MIRS (MMX), ExoMars

- **Ground instrumentation**

Atacama Large subMillimeter Array (ALMA + WSU 2030), SPIAKID



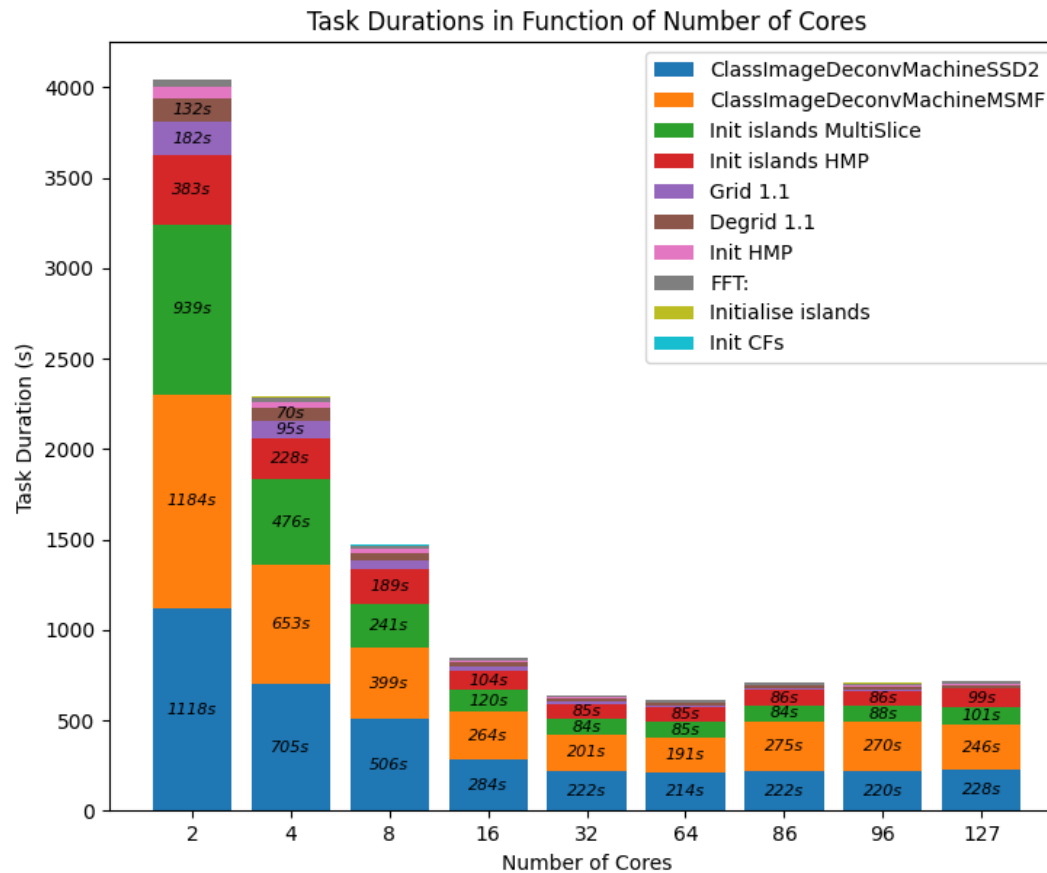
Example of execution on the ALCOR server



Profiling script (python)



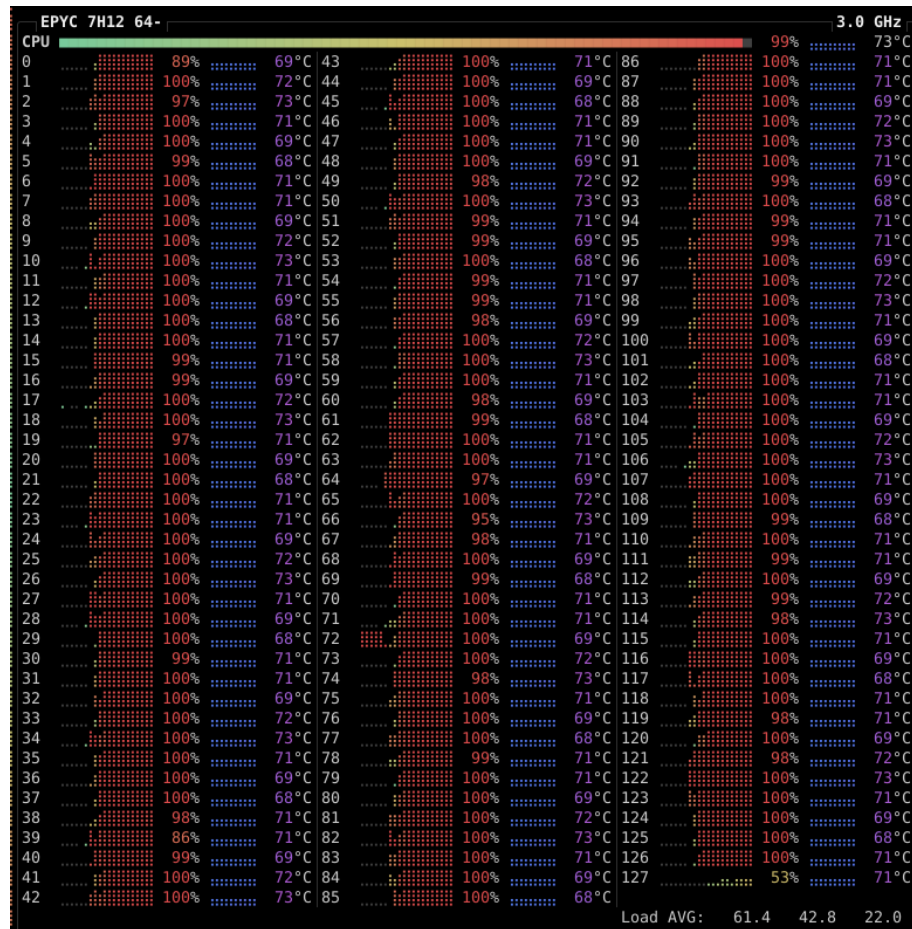
- Use of DDFacet logs to profile the execution time of the various stages
- Performance degradation > 64 cores: NUMEXPR_MAX_THREADS variable?
- Evolution of the duration of the various DDFacet stages as a function of the number of cores



Profiling with Btop

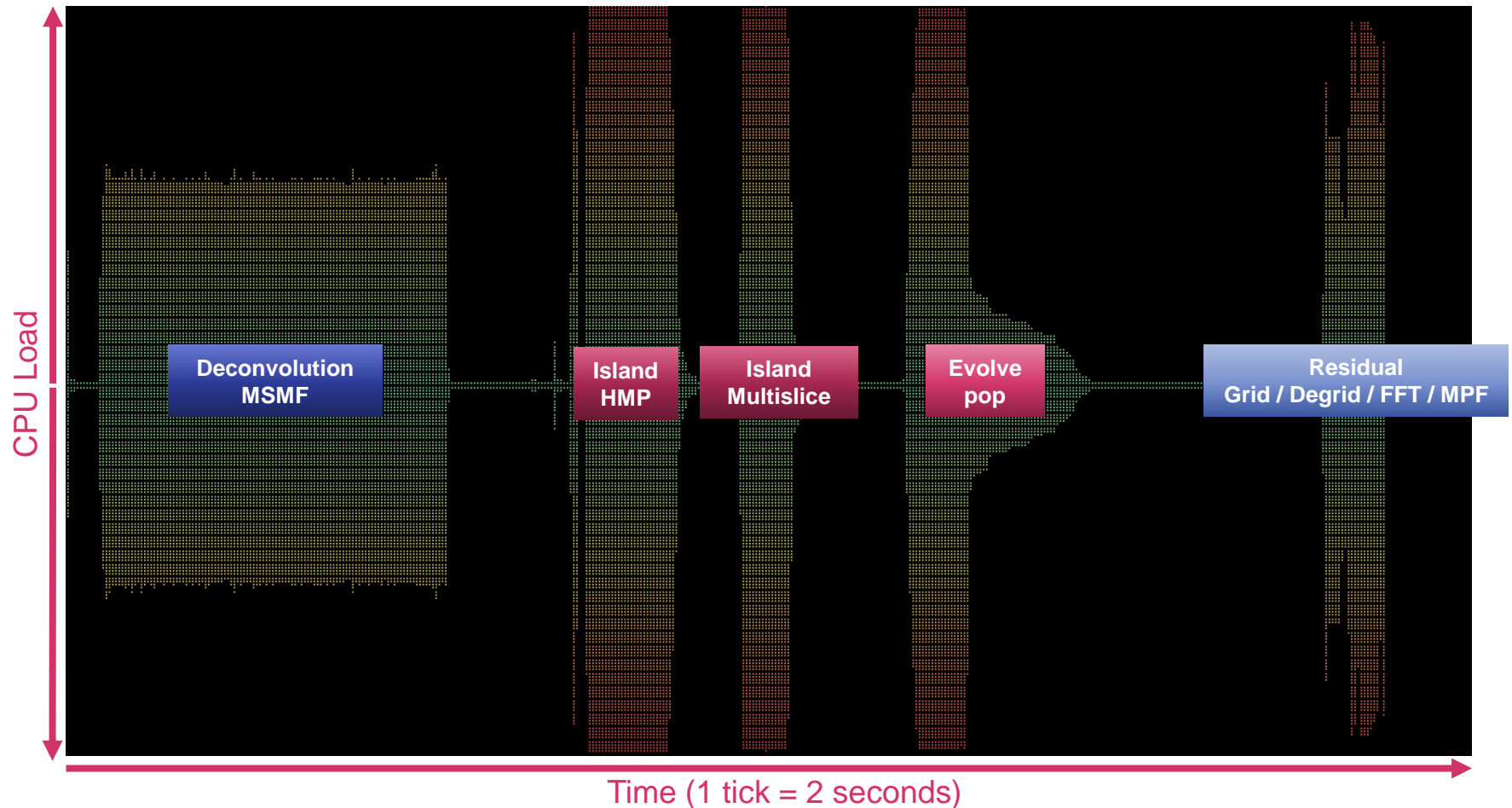


- Exploitation of the 128 logical cores of the CPU during the execution on ALCOR (LAB)
- CPU : AMD EPYC 7H12 64-Core Processor
- Profiling : Btop++ (<https://github.com/aristocratos/btop>)



Profiling with Btop

- CPU core utilisation over time
- Labelling the different stages of the DDFacet Pipeline



Profiling with Perf



- CPU consumption broken down by low-level function call, depending on DDF current stage

```
Terminal - vhazard@alcor: ~/DDFacet/spack
Samples: 82M of event 'cycles', 4000 Hz, Event count (approx.): 2429246721495 lost: 0/224143 drop: 0/307433
Overhead Shared Object Symbol
54.66% [kernel] GOMP_taskloop_ull
15.00% _pyArrays3x.so [.] 0x000000000001b96
10.10% _pyArrays3x.so [.] 0x000000000001b90
8.18% _pyArrays3x.so [.] 0x000000000001b83
8.17% _pyArrays3x.so [.] 0x000000000001b80
0.32% perf [.] queue_event
0.11% [unknown] [.] 0x00007ff25e551d34
0.09% perf [.] deliver_event
0.06% perf [.] ordered_events_flush.part.0
0.06% [unknown]
0.06% libc.so.6 [.]
0.05% [kernel]
0.05% [unknown]
0.04% [unknown]
0.04% [unknown]
0.04% perf [.]
0.04% perf [.]
0.03% [kernel] [k] update_cls_group
0.03% [kernel] [k] native_queued_spin_lock_slowpath.part.0
0.03% perf [.] evsel_parse_sample
0.03% perf [.] _histfs_add_entry_constprop.0
0.03% libQt5Gui.so.5.15.14 [.] png_write_find_filter
0.03% [kernel] [k] update_sg_lb_stats
0.03% perf [.] ordered_events_queue
0.03% perf [.] rb_next
0.02% [unknown] [.] 0x00007ff25ee841c
0.02% _pyArrays3x.so [.] 0x000000000001b7b
```

For a higher level overview, try: perf top --sort comm,dso

Deconvolution
MSMF

```
Terminal - vhazard@alcor: ~/DDFacet/spack
Samples: 108M of event 'cycles', 4000 Hz, Event count (approx.): 2810551477151 lost: 100426/18687494 drop: 1
Overhead Shared Object Symbol
25.13% interpreter.cpython-39-x86_64-linux-gnu.so [.] 0x000000000000bb575
3.24% interpreter.cpython-39-x86_64-linux-gnu.so [.] 0x0000000000024053
3.24% interpreter.cpython-39-x86_64-linux-gnu.so [.] 0x000000000002400a
3.24% interpreter.cpython-39-x86_64-linux-gnu.so [.] 0x000000000002406d
3.10% interpreter.cpython-39-x86_64-linux-gnu.so [.] 0x0000000000024088
2.13% interpreter.cpython-39-x86_64-linux-gnu.so [.] 0x000000000000bbdf
1.69% interpreter.cpython-39-x86_64-linux-gnu.so [.] 0x0000000000024005
1.66% interpreter.cpython-39-x86_64-linux-gnu.so [.] 0x0000000000024029
0.92% [unknown]
0.93% [unknown]
0.75% [unknown]
0.73% [unknown]
0.66% [unknown]
0.64% interpreter.cp
0.59% interpreter.cp
0.48% [unknown]
0.48% perf [.]
0.45% [unknown] [.] 0x00007ff25e55266e
0.45% [unknown] [.] 0x00007ff25e55266a
0.43% interpreter.cpython-39-x86_64-linux-gnu.so [.] 0x000000000000bbdb4
0.42% interpreter.cpython-39-x86_64-linux-gnu.so [.] 0x0000000000024076
0.41% [unknown] [.] 0x00007ff25e4e6313
0.41% [unknown] [.] 0x00007ff25e52a3ec
0.41% interpreter.cpython-39-x86_64-linux-gnu.so [.] 0x000000000002405c
0.40% interpreter.cpython-39-x86_64-linux-gnu.so [.] 0x0000000000024091
0.39% [unknown] [.] 0x00007ff25e4e84dd
0.35% interpreter.cpython-39-x86_64-linux-gnu.so [.] 0x000000000002404e
```

Island HMP

```
Terminal - vhazard@alcor: ~/DDFacet/ddfacet-test
Samples: 128M of event 'cycles', 4000 Hz, Event count (approx.): 200563693426 lost: 0/3317811
Overhead Shared Object Symbol
5.60% libc-2.31.so [.] 0x0000000000160503
5.60% libc-2.31.so [.] 0x000000000016050d
5.22% libc-2.31.so [.] 0x0000000000160508
4.89% libopenblas-r0-09e95953.3.13.so [.] 0x0000000000f54100
4.83% libc-2.31.so [.] 0x00000000001604fe
4.24% libopenblas-r0-09e95953.3.13.so [.] 0x0000000000f540be
3.84% libopenblas-r0-09e95953.3.13.so [.] 0x0000000000f540e4
3.24% libopenblas-r0-09e95953.3.13.so [.] 0x0000000000f540ce
2.95% libc-2.31.so [.] 0x0000000000160520
2.73% libc-2.31.so [.] 0x000000000016052c
2.12% libopenblas-r0 [.] 0x0000000000f540f9
2.06% libopenblas-r0 [.] 0x0000000000f540b8
1.97% libc-2.31.so [.] 0x0000000000160532
1.93% libc-2.31.so [.] 0x0000000000160526
1.73% libopenblas-r0-09e95953.3.13.so [.] 0x0000000000f540d5
1.21% libopenblas-r0-09e95953.3.13.so [.] 0x0000000000f540ea
1.10% perf [.] rb_next
0.80% perf [.] hpp_sort_overhead
0.53% multiarray_umath.cpython-39-x86_64-linux-gnu.so [.] 0x000000000025c152
0.39% libjvm.so [.] ConnectionGraph::add_fiel
0.35% libopenblas-r0-09e95953.3.13.so [.] 0x0000000000f54048
0.32% libc-2.31.so [.] 0x00000000001604d0
0.31% libopenblas-r0-09e95953.3.13.so [.] 0x0000000000f540b2
0.29% libc-2.31.so [.] 0x00000000001604bb
```

Evolve pop

```
Terminal - vhazard@alcor: ~/DDFacet/spack
Samples: 31M of event 'cycles', 4000 Hz, Event count (approx.): 88127877174 lost: 0/2241
Overhead Shared Object Symbol
62.85% [kernel] native_queued_spin_lo
1.41% [unknown] [.] 0x00007ff25e552464
0.74% [unknown] [.] 0x00007ff25e630d3d
0.71% [unknown] [.] 0x00007ff25e551d34
0.63% [kernel] [k] asm_exc_page_fault
0.59% [unknown] [.] 0x00007ff25eea858b
0.52% [kernel] [k] pageblock_pfn_to_pa
0.45% [unknown] [.] 0x00007ff25eea8617
0.39% [unknown] [.] 0x00007ff25e551fb8
0.38% [unknown] [.] 0x00007ff25eea8621
0.35% [unknown] [.] 0x00007ff25e4f1e6f
0.34% [unknown] [.] 0x00007ff25e551f58
0.31% [unknown] [.] 0x00007ff25e62f566
0.31% [unknown] [.] 0x00007ff25e551f54
0.27% [unknown] [.] 0x00007ff25e62f4f8
0.26% perf [.] queue_event
0.25% [unknown] [.] 0x00007ff25eea8595
0.24% [unknown] [.] 0x00007ff25e62f562
0.23% [unknown] [.] 0x00007ff25e551fb4
0.22% [unknown] [.] 0x00007ff25e62f4f4
0.21% libxrdp.so.0.0.0 [.] compress_rdp
0.21% libfftw3f.so.3.5.8 [.] 0x000000000002a7a1
0.21% [unknown] [.] 0x00007ff25eea859a
0.20% [unknown] [.] 0x00007ff25e551d3b
```

Residual
Grid / Degrid /
FFT / MPF

Profiling with BenchMonSPC



Profiling CPU Load, Memory, Disk, & Files

BenchMonSPC

Auteurs :
Anass Serhani
Shan Mignot

Data set :

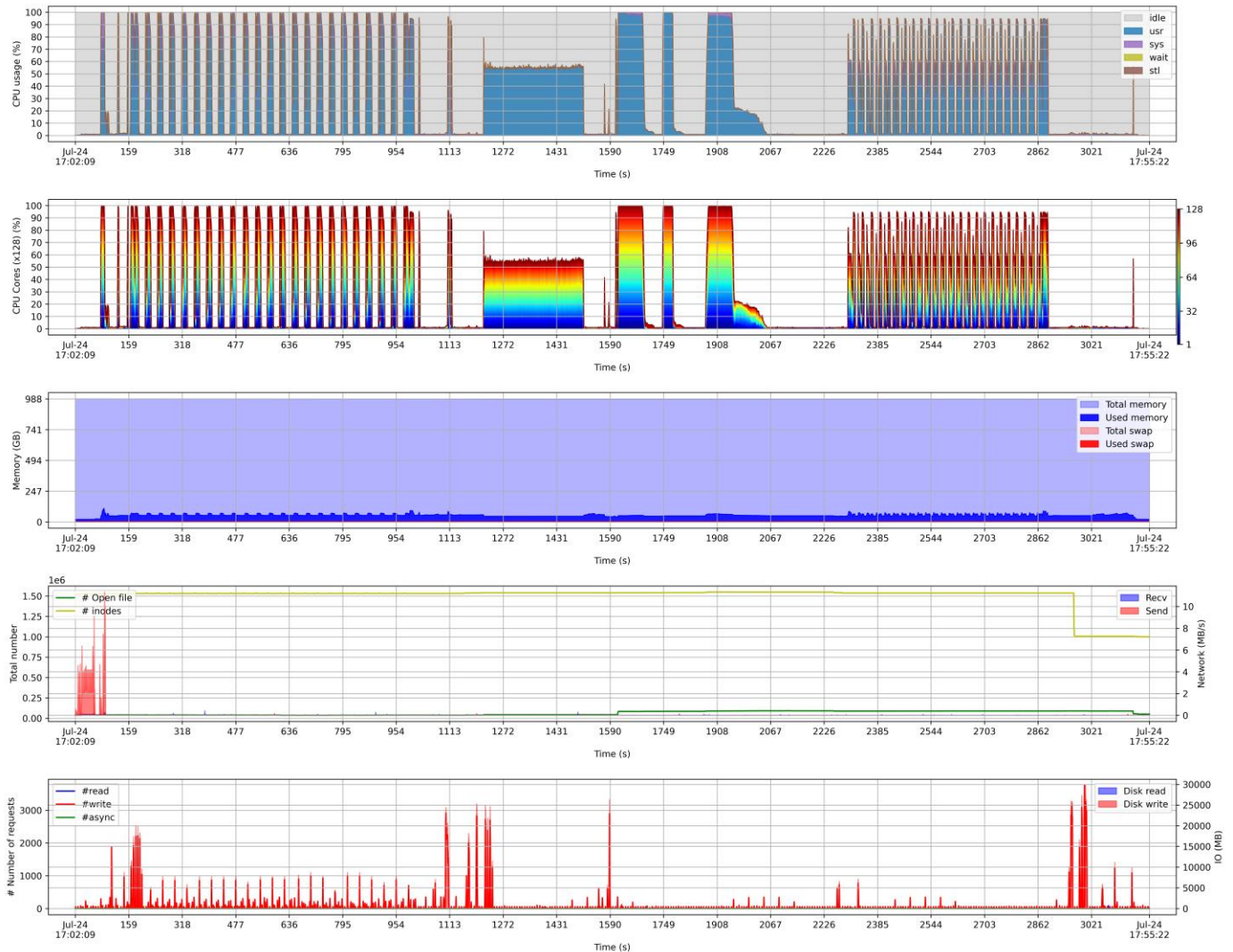
24MS from LOFAR
L526161 - SB244

Server :

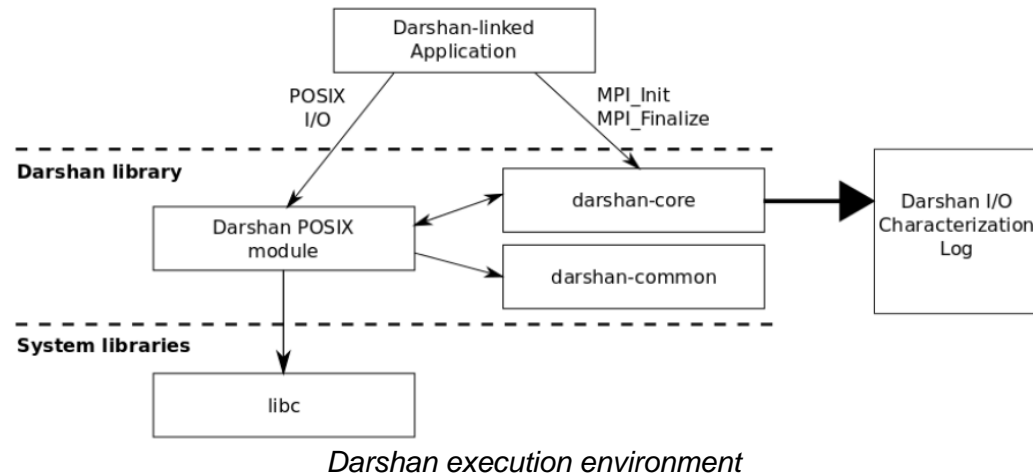
ALCOR (DAS06)

DDFacet

50-minute run

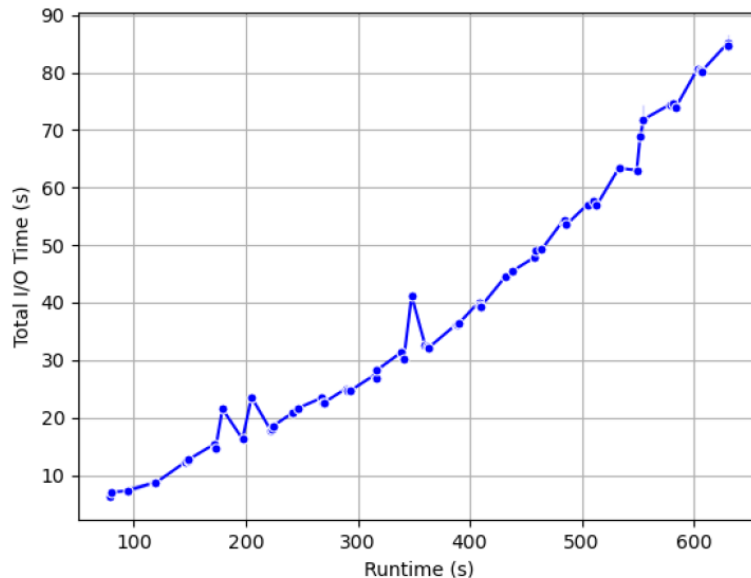


- Iheb Becher (M2 internship at LAB & INRIA) worked to use *Darshan* and profile IOs with a trace report, using dynamic instrumentation
- Darshan's *libdarshan* library intercepts system function calls, relying on two main components: *darshan-core* and *darshan-common*.



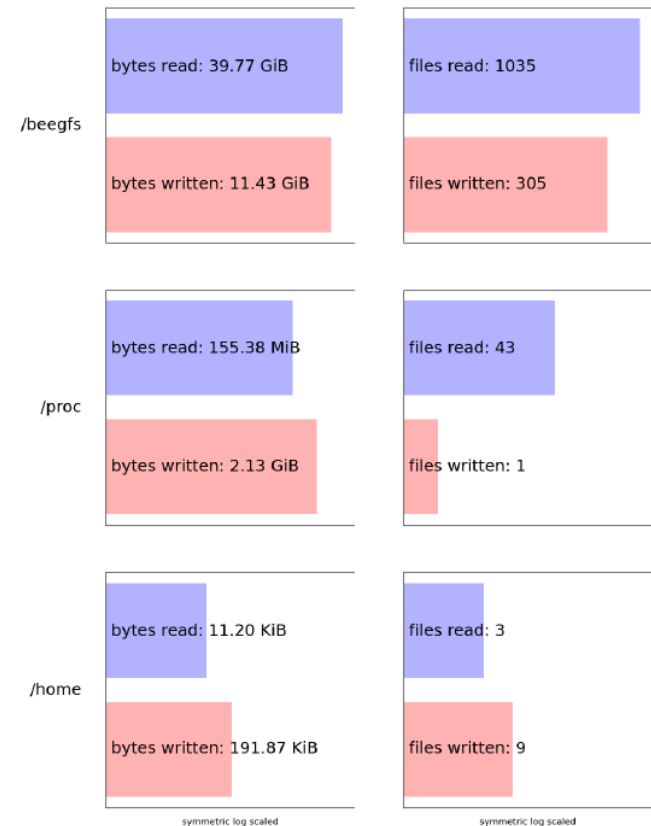
- Darshan Python API (PyDarshan):
 - Read Darshan trace files (.darshan)
 - Accessing metadata
 - Exploration of I/O modules

- Preliminary I/O profiling results on a single DDFacet run (10-minute execution)



I/O time of DDFacet relative to total execution time

- Note: this doesn't necessarily mean there's an I/O bottleneck, as DDFacet executes asynchronously
- Darshan fails to capture all write operations by DDFacet, as it underestimates the total cache file size (e.g. 51.64 GB vs. 11.43 GB) measured by a system script



Data access model by file system

- Several deployment solutions have been tested so far for the DDF pipeline, as an alternative to the use of Singularity containers (Spack & Guix)
- The main challenges are deploying code to ensure reproducible environments and optimizing execution performance on the target machine



Spack



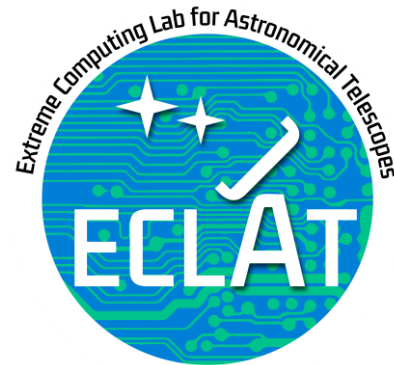
Guix

- ⌚ **Spack** deployment with the help of David Guibert (Eviden). This development is at standstill, error compiling sources (github.com/dguibert/spack/tree/dg/ddfacet-busy-week)
- ✅ Development of **Guix** package to deploy DDFacet, work carried out by Olivier Aumage (INRIA - STORM). Functional and deployed on PlaFRIM and ALCOR

What do to next?



- Coordinate efforts between different labs & industry within ECLAT
- New HR: contract hire to join the CSSD team at LAB for 2025 to work on this particular topic (as part of NumPEX)
- There's still work to be done on software deployment, on both DDF and ICAL pipelines
- Understanding the algorithms and orchestration of imaging software remains a weak spot





Spack

- Python packages
- Does not require root to build
- Does dependency resolution
- Binaries are relocatable



Guix

- Lightweight
- Easy to deploy with root access
- More precise descriptions for large environments