COSMIC: AN ADPTABLE, POWERFUL AND PROVEN RTC PLATFORM

Damien Gratadour

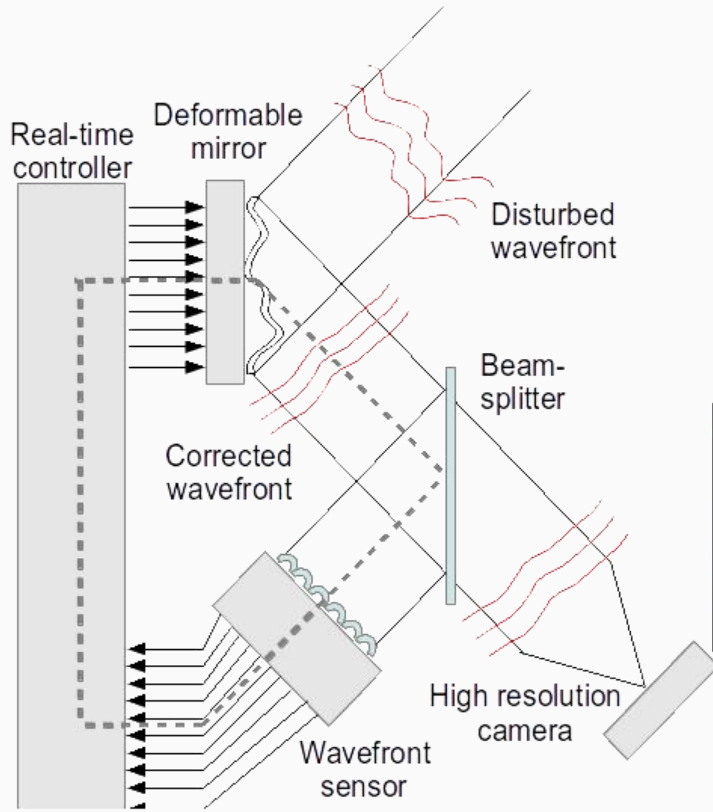
Acknowledgments

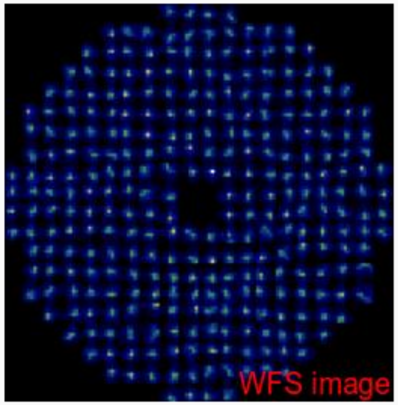
- Observatoire de Paris - Université PSL
  - F. Ferreira, A. Sevin, J. Plante and C. Cetre
- Research School of Astronomy & Astrophysics, ANU
  - J. Smith, C. Gretton, N. Doucet, J. Cranney, J. Bernard and F. Rigaut
- Extreme Computing Research Center, KAUST
  - H. Ltaief, Y. Hong & D. Keyes
- Innovative Computing Laboratory, UTK
  - Q. Cao, Y. Pei, G. Bosilca, J. Dongarra
- NVIDIA
  - S. Jones and I. Said
- Barcelona Supercomputing Center & UPC
  - B. Pou Mulet, E. Quinones & M. Martin

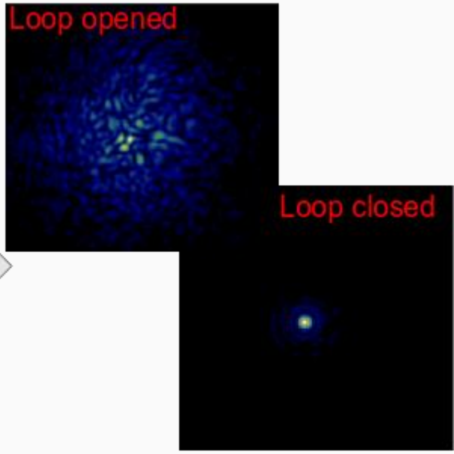
Adaptive Optics for giant telescopes

Control in real-time the shape of the incoming wavefront

- **Sensors are cameras** equipped with an optical device (lenslet array, pyramidal prism, etc…)
- **Deformable mirrors** to compensate for wavefront distortions
- **Typical rate of operation is 1kHz**
- Compute **pipeline latency below 1 millisecond**
- **Stable time-to-solution** is critical to ensure stable Operations (jitter of the order of 10s of µs)





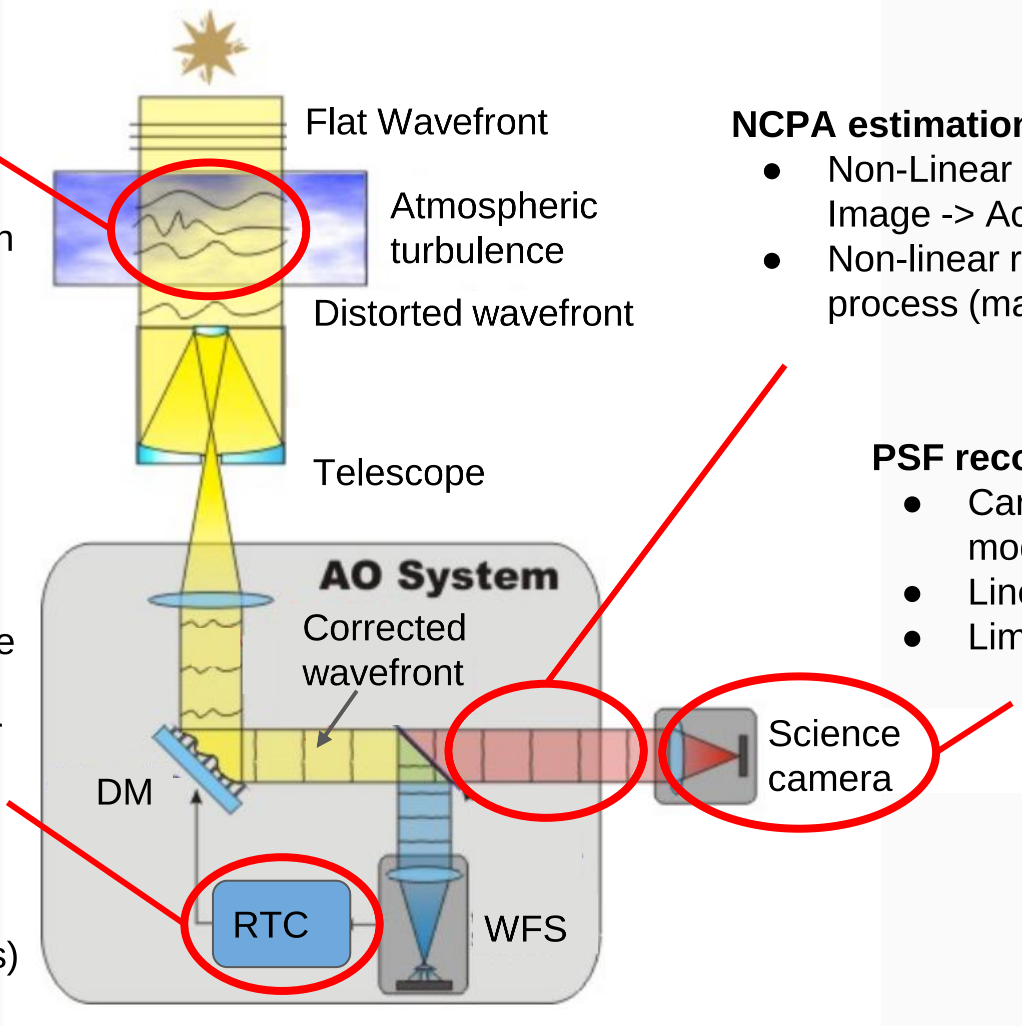


"Classical ML" (or Model Driven ML) for AO

AO is this kind of instrumentation that involves a lot of numerical tools

Turbulence Profiling
- Complex model turbulence -> measurements
- Iterative reconstruction process (cost function optimization)

Wavefront Reconstruction:
- Linear relationship WFS -> Actuators
- Results from turbulence profiling used to built the linear reconstructor (regularization term)
- Non-linear approaches to avoid the burden of dense linear algebra (but same assumptions)



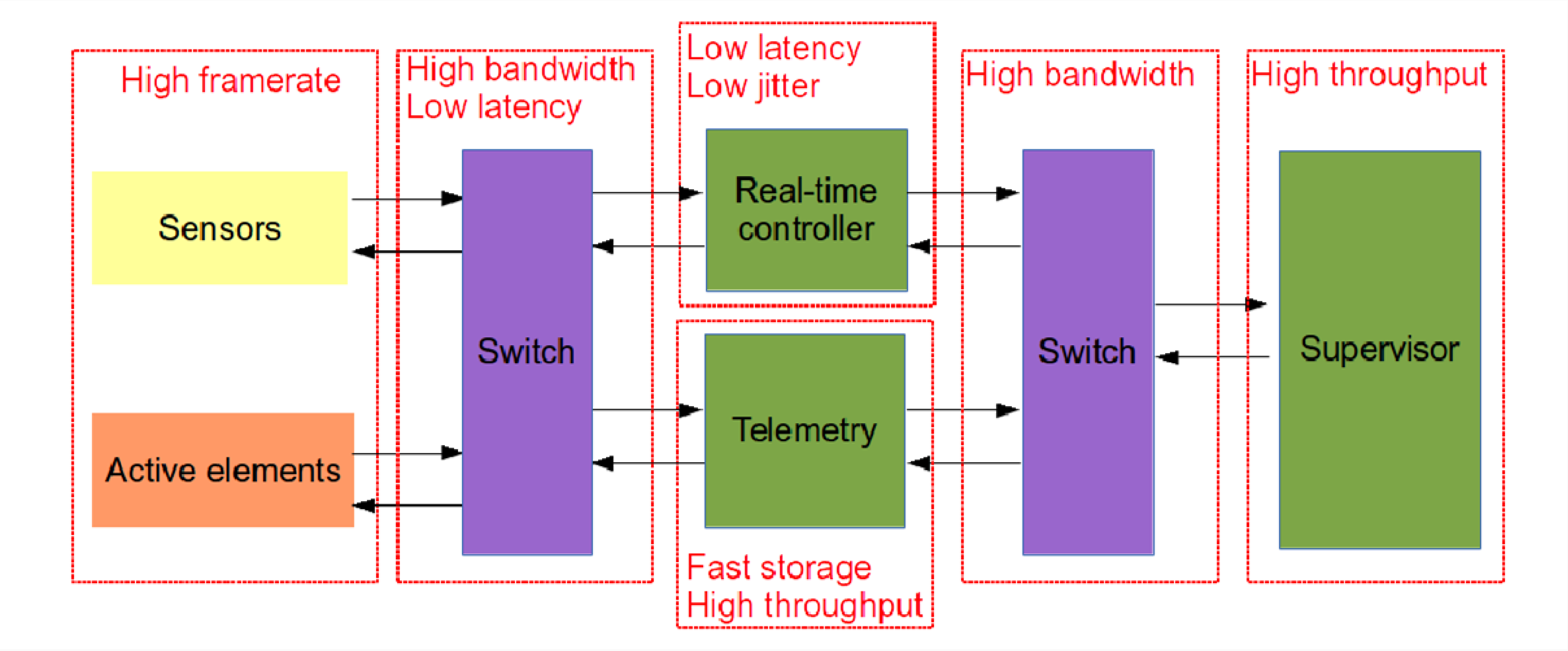
NCPA estimation
- Non-Linear relationship Image -> Actuators
- Non-linear reconstruction process (max likelihood)

PSF reconstruction
- Careful error budget modeling
- Linear reconstruction
- Limited to long exposure

AO RTC Global system architecture

Based on heterogeneous architecture to implement main functions

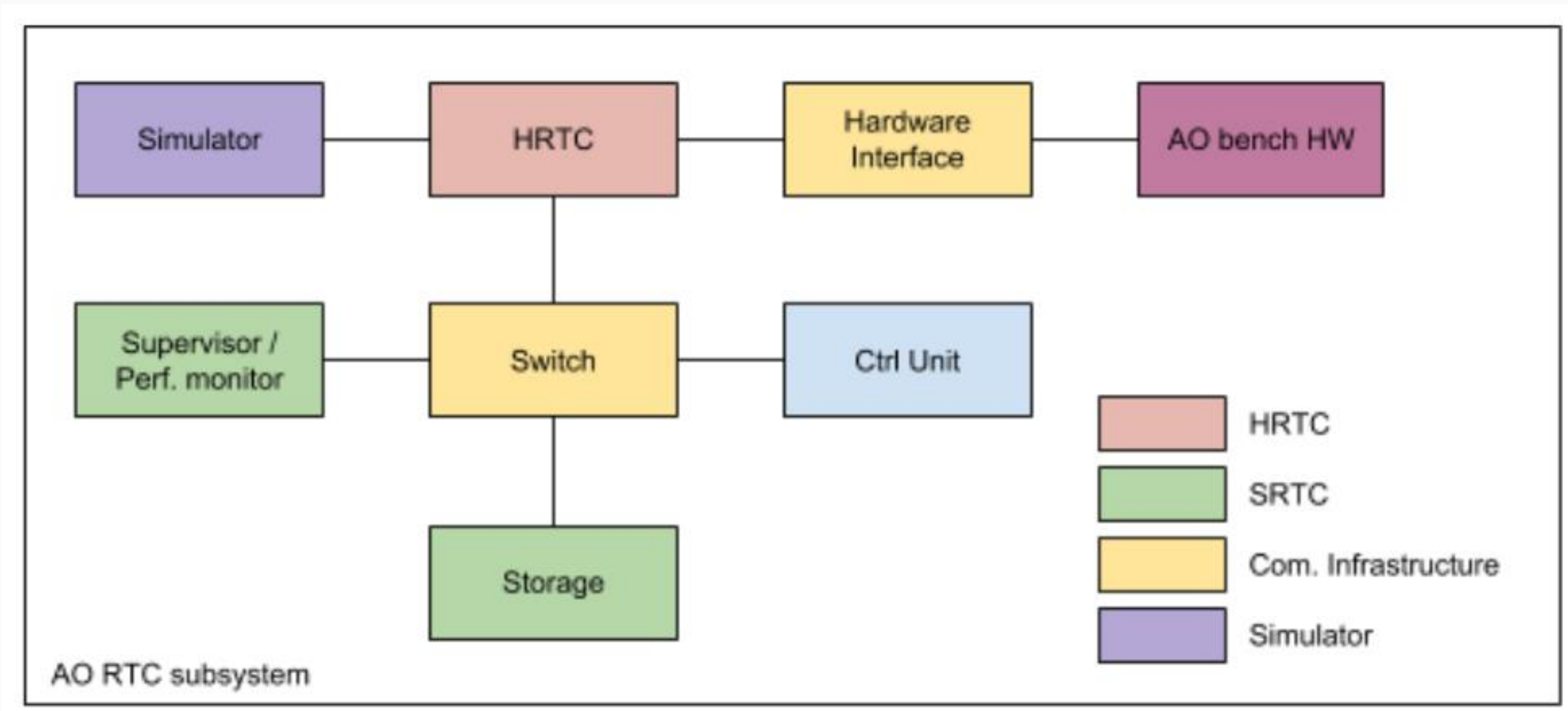
- Cope with various functional & non-functional requirements for the different sub-systems
- Mix high throughput Machine Learning (supervisor, a.k.a. SRTC) with low latency & low jitter computing (real-time controller, a.k.a. HRTC)



AO RTC Hardware components

Typical functional decomposition

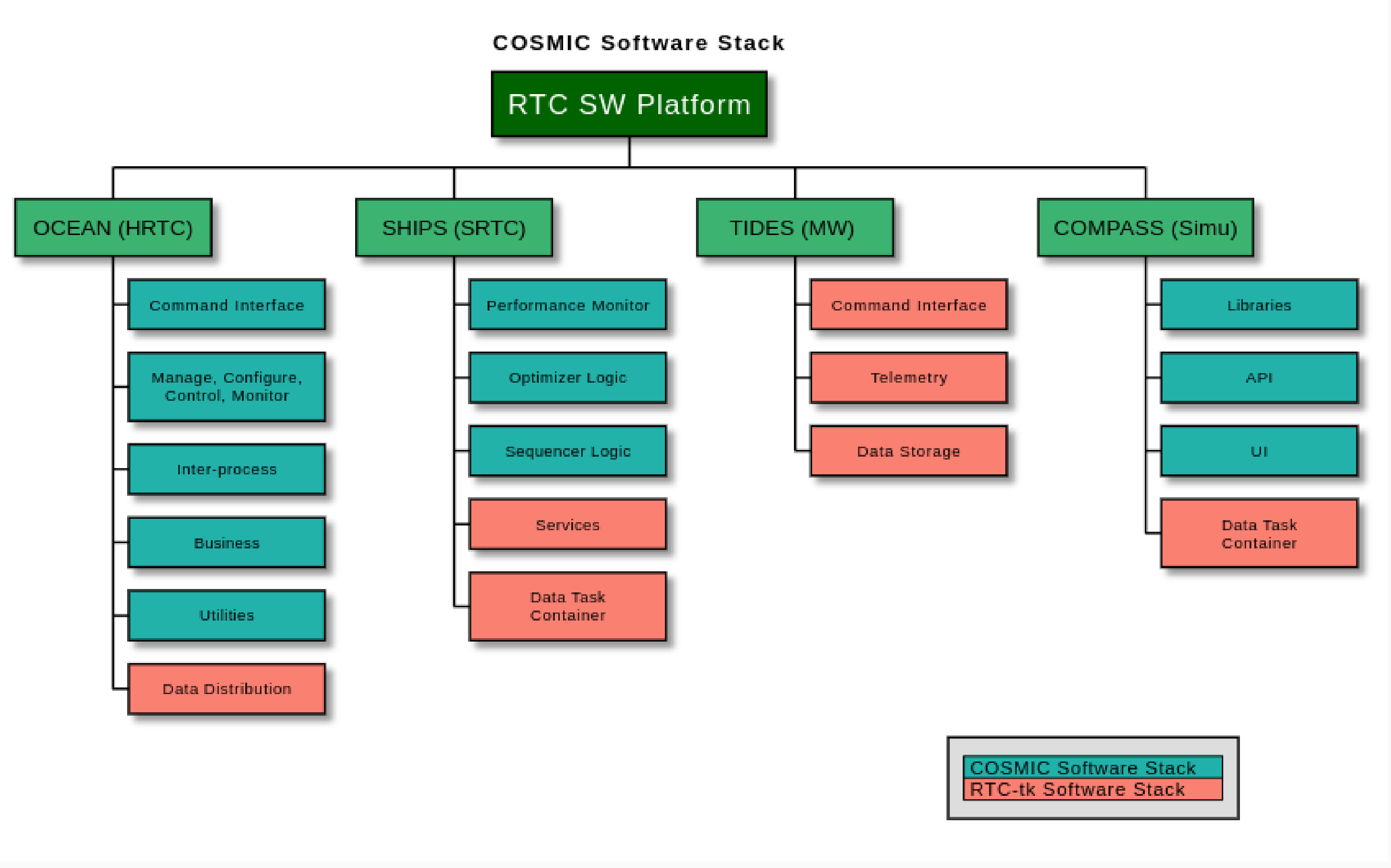
- Mostly aligned with SPARTA / ESO specifications
- Including simulator sub-system



AO RTC Software components

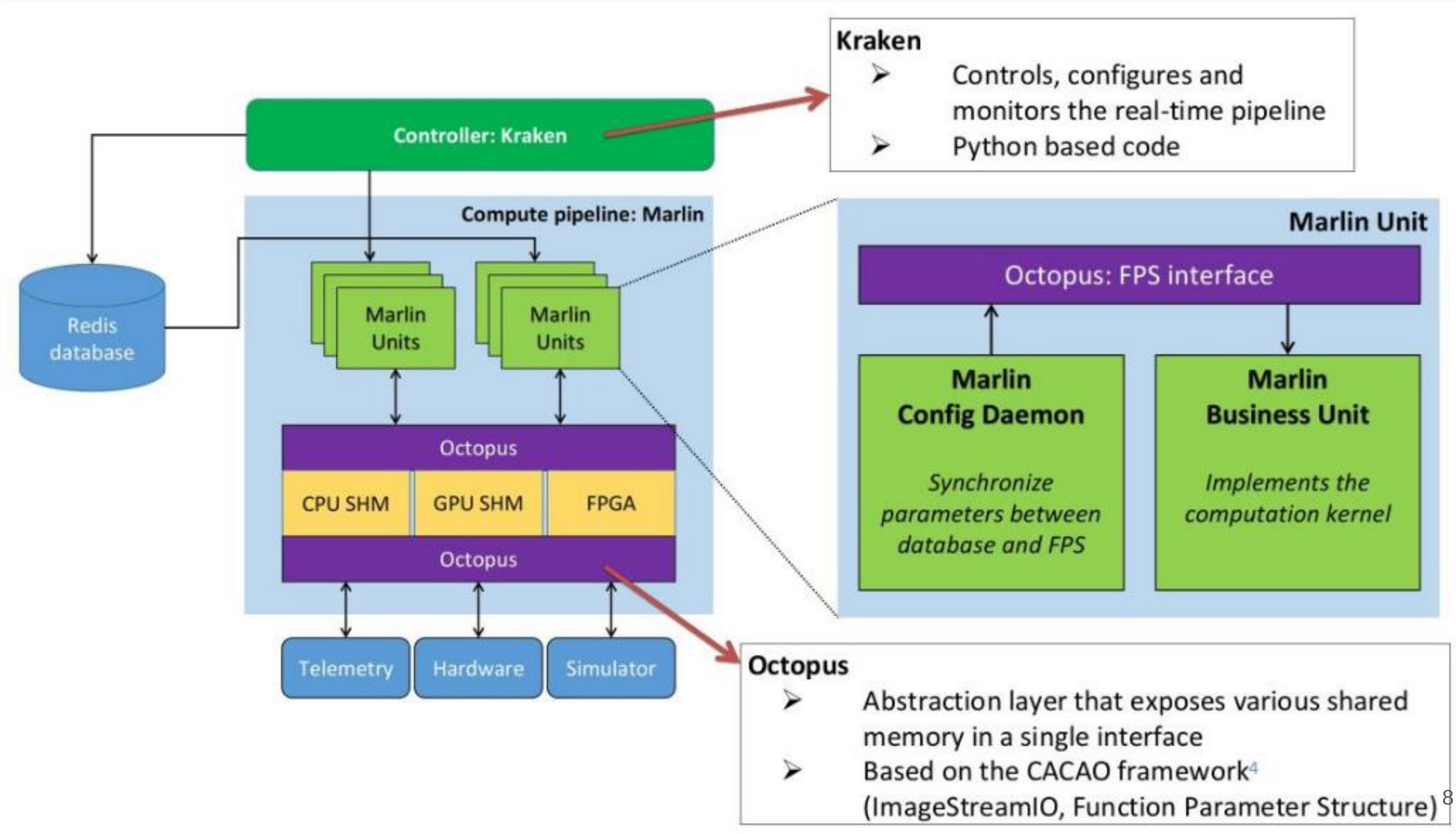
4 components:

- OCEAN, SHIPS, TIDES and COMPASS
- Integration with ESO's RTC Toolkit



HRTC SW DESIGN

Core pipeline, made highly modular through the Marlin library



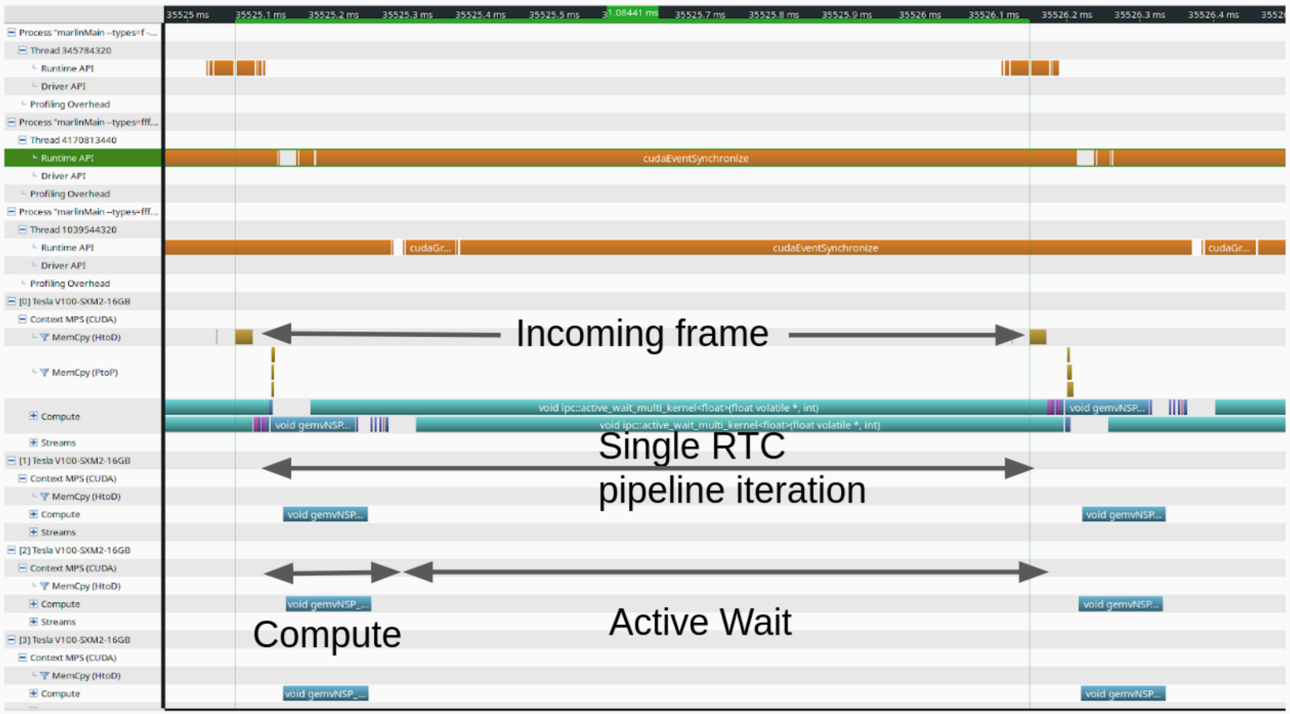
Core features

- Python interface: user-friendly RTC, **configuration / execution made easy**
- Multiple processes interacting in real-time: BUs encapsulating kernels
  - **Any kind of kernels, including standard libraries: BLAS, FFT** (highly portable)
- **Same interface across the pipeline**: RT data streams, Configuration parameters (FPS), Telemetry (handled by independent processes)
  - But different shared memory domains !
- Trade-off between **modularity and efficiency**: BUs can be grouped into **containers** to reproduce main functions while integrating sub-functions:
  - Example of WFS data processing: include pixel processing + centroiding and work on multiple WFS
  - Each sub-function is a BU (can be debugged / tested individually) all regrouped in a single container (i.e. uses a single process)
  - Stream programming provides **concurrent kernels execution when needed**

TYPICAL HRTC PERFORMANCE

Representative pipeline including frame transfer + centroiding + MVM

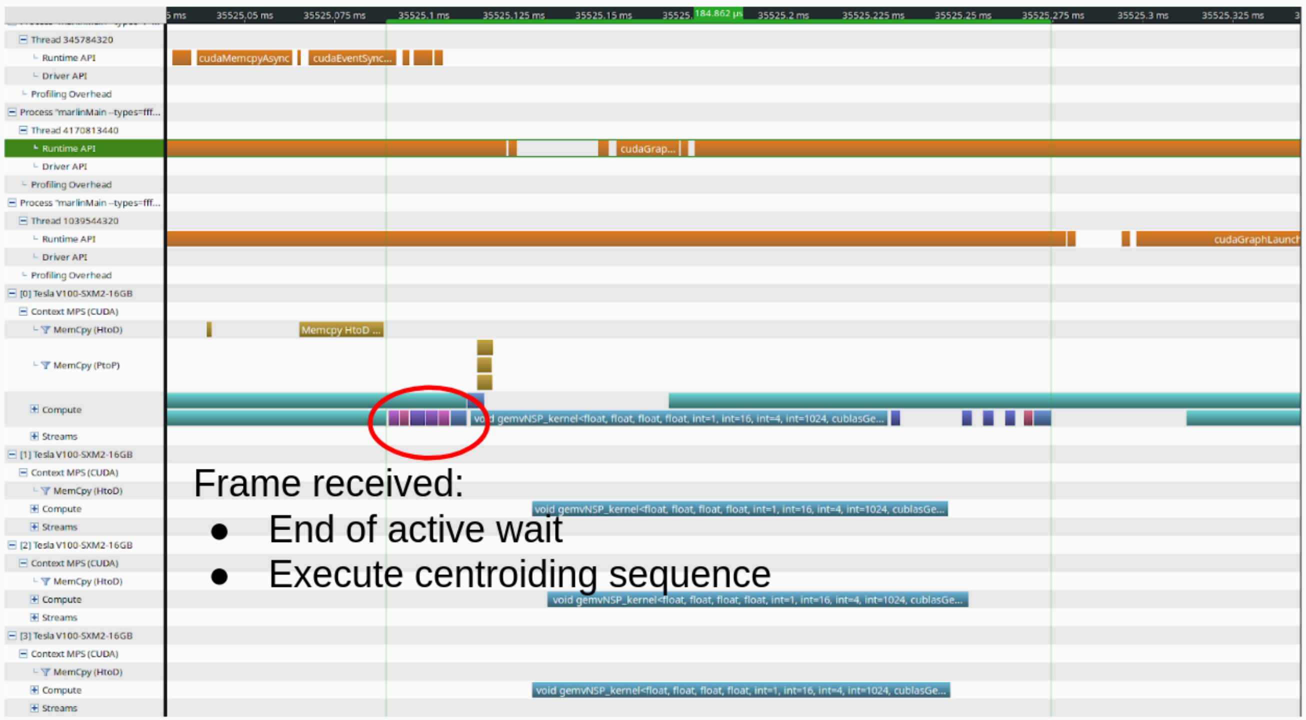
- WFS frames are sent by a hardware emulator at a regular rate (1 kHz)
- GPU is mostly “actively waiting”



Typical HRTC performance

Representative pipeline including frame transfer + centroiding + MVM

- WFS frames are sent by a hardware emulator at a regular rate (1 kHz)
- GPU is mostly "actively waiting"



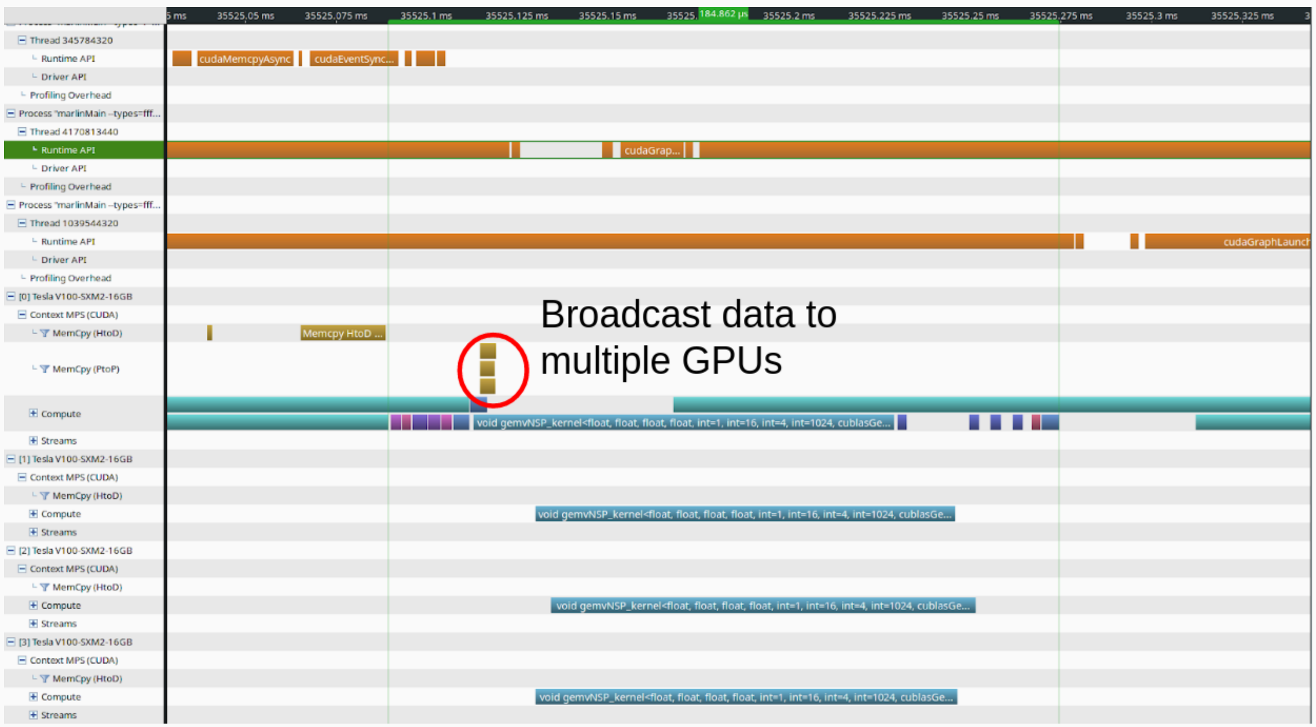
Frame received:

- End of active wait
- Execute centroiding sequence

Typical HRTC performance

Representative pipeline including frame transfer + centroiding + MVM

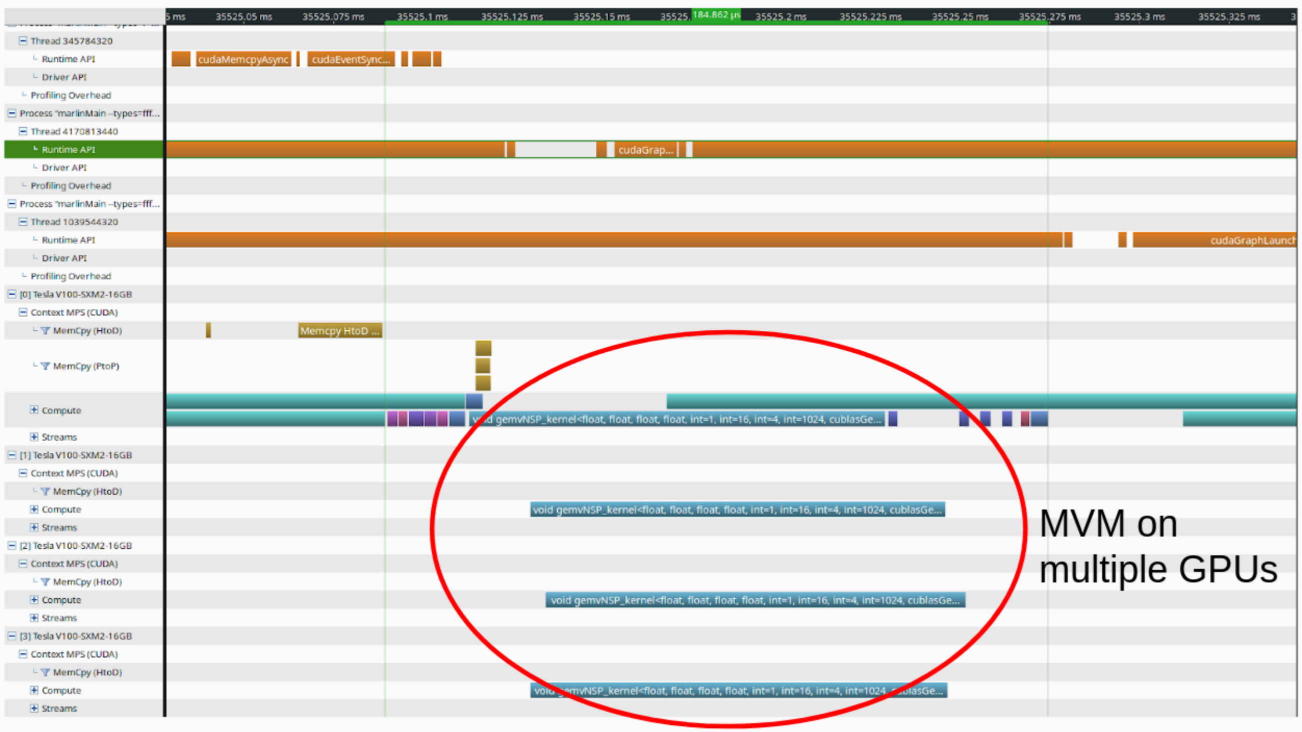
- WFS frames are sent by a hardware emulator at a regular rate (1 kHz)
- GPU is mostly “actively waiting”



Typical HRTC performance

Representative pipeline including frame transfer + centroiding + MVM

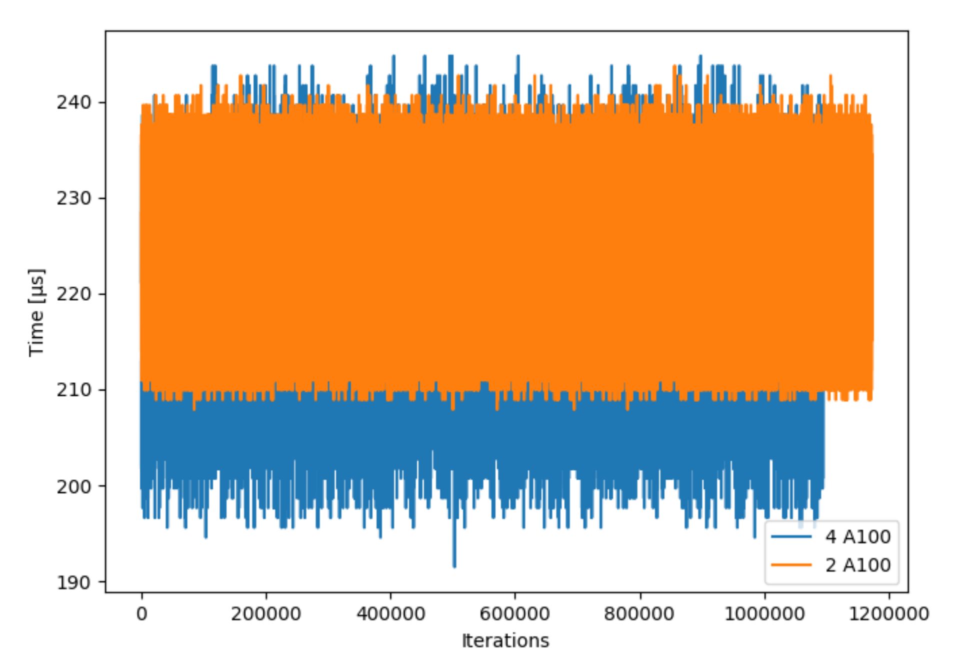
- WFS frames are sent by a hardware emulator at a regular rate (1 kHz)
- GPU is mostly “actively waiting”

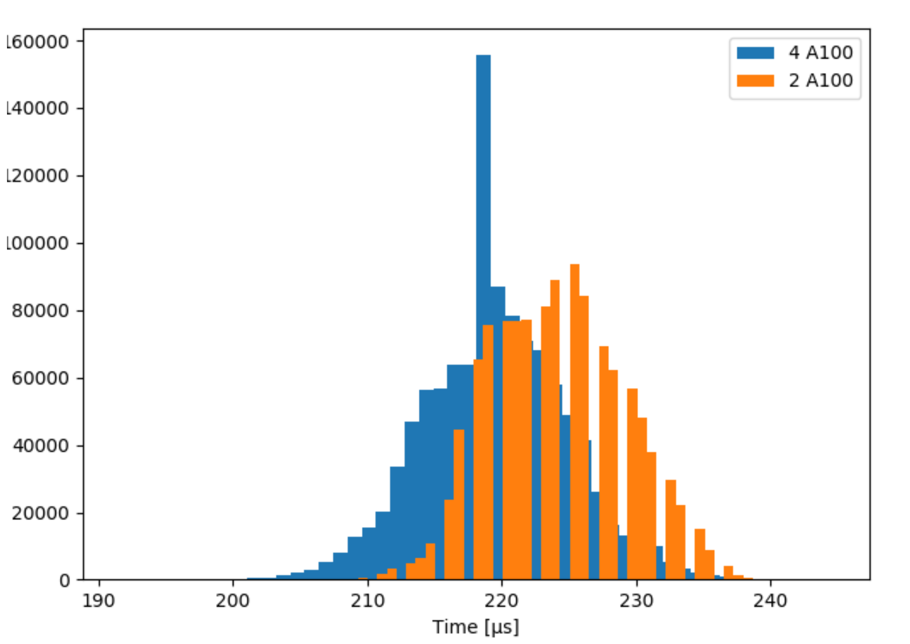


Typical HRTC performance

Comparison of end-to-end latency on several GPU generations

- MAVIS case (5k x 20k command matrix)
- Mostly memory bound: benefit from increased mem. bandwidth between V100 and A100
- Almost perfect scaling !





Adaptable, powerful ... and proven RTC platform

Facility instruments

- **Keck**: already online, delivering science (**see R. Biasi's talk**)
- **Micado**: being integrated (**see F. Ferreira's talk**)
- **MAVIS**: final design phase, passed preliminary design
    - Global design and results: **see F. Rigaut's talk**
    - SRTC architecture & benchmarking: **see N. Doucet's talk**
- **SPHERE+**: preliminary design phase
- (**NenuFAR**): important building blocks (e.g. data ingestion) tested and integrated on radio-telescope for transients detection (**see J. Plante's talk**)

Lab experiments

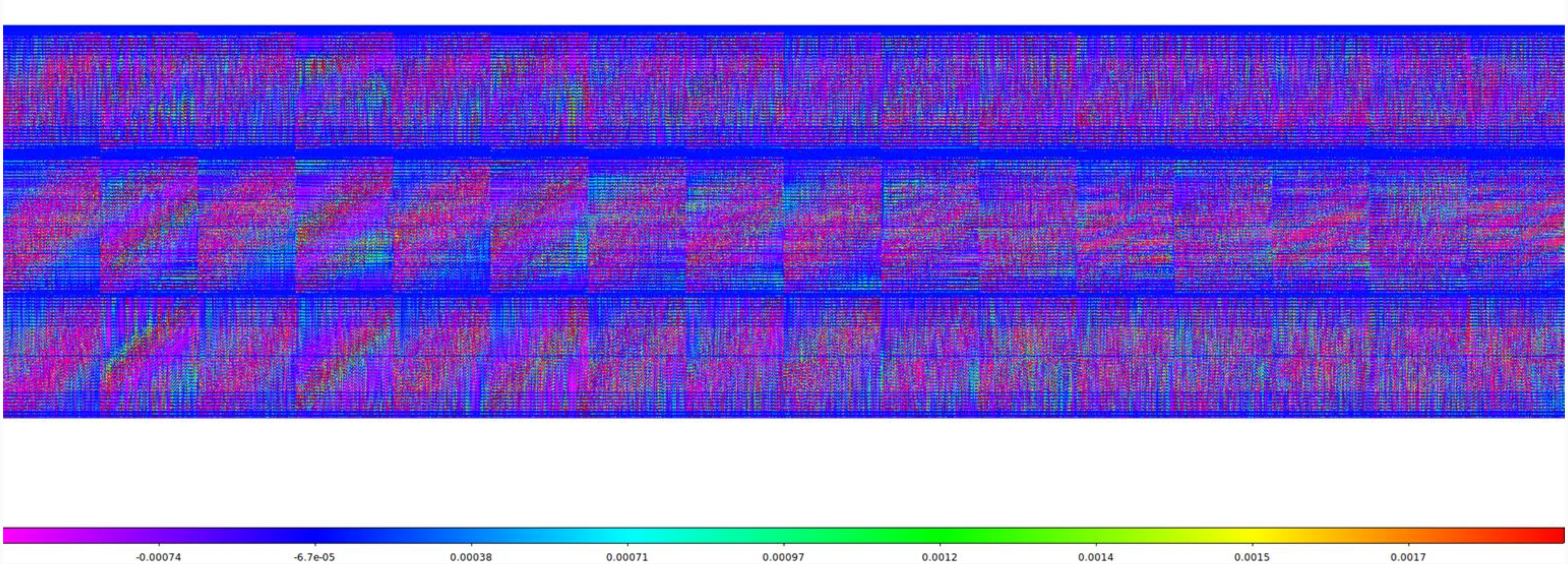
- GHOST @ ESO: used to drive the AO bench and interface with ML (see Jalo's talk)
- LabRTC @ INAF: used for prototyping (incl. on-sky) new WFS concepts
- Micado demo @ LESIA (up and running at scale, see Florian's talk)

Adaptable, Powerful … and Future-Proof!

Improving performance portability

A closer look at the tomographic reconstructor

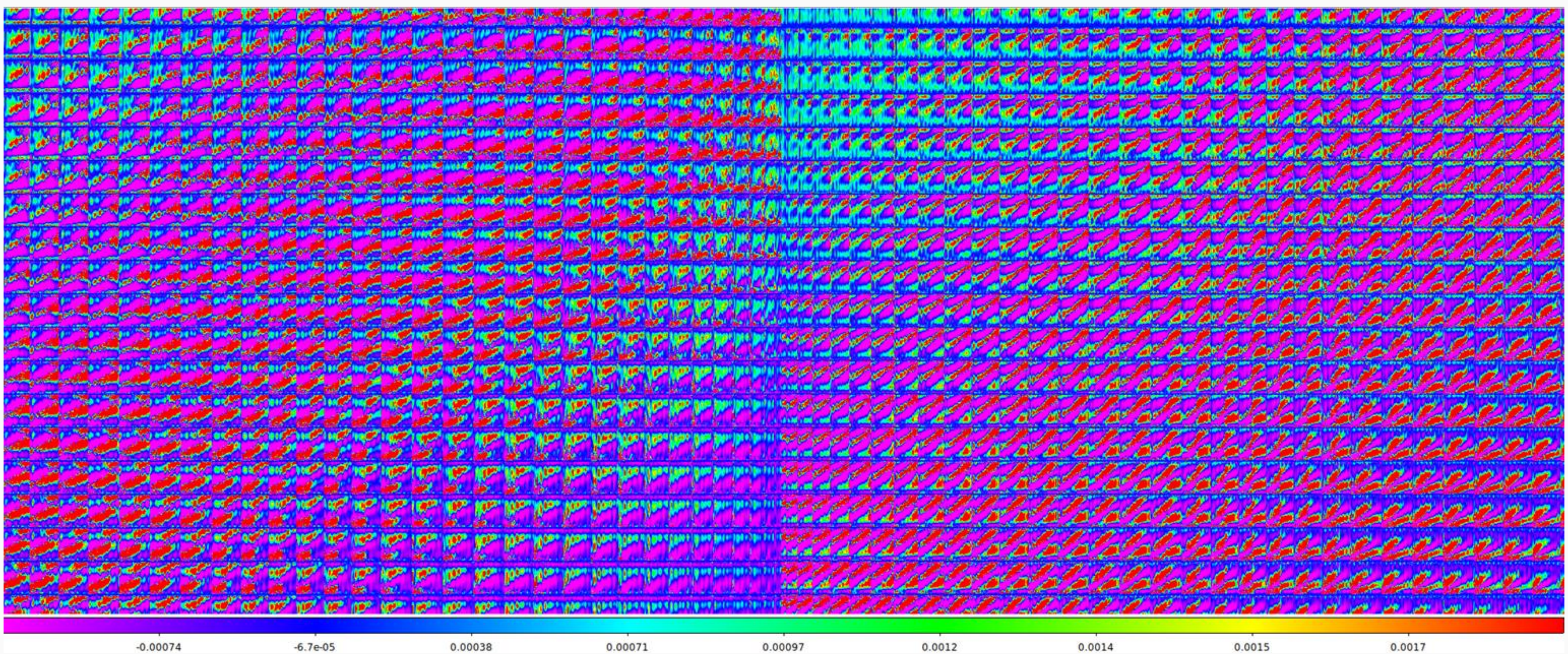
- Tomography + predictive control
- Apparently very structured:
  - can be connected to system parameters (WFS dimensioning)
  - Very low structural dependency wrt turbulence parameters



Improving performance portability

A closer look at the tomographic reconstructor

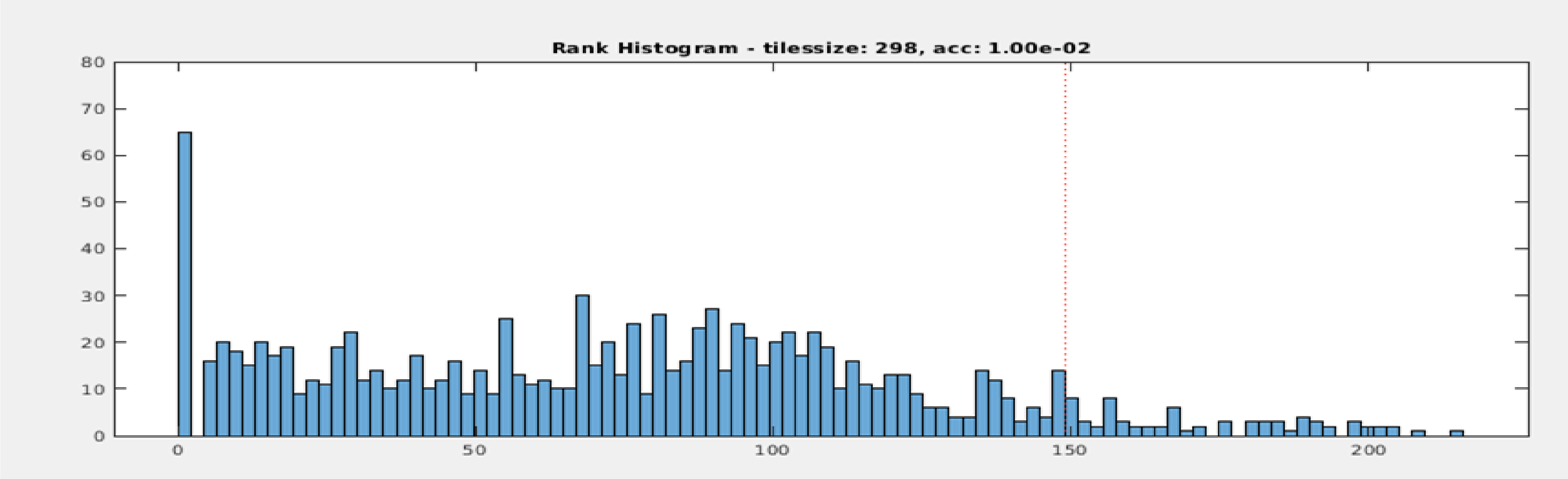
- Tomography + predictive control
- Apparently very structured:
  - can be connected to system parameters (WFS dimensioning)
  - Very low structural dependency wrt turbulence parameters



Improving performance portability

Ranks analysis: splitting the matrix into tiles and looking at ranks

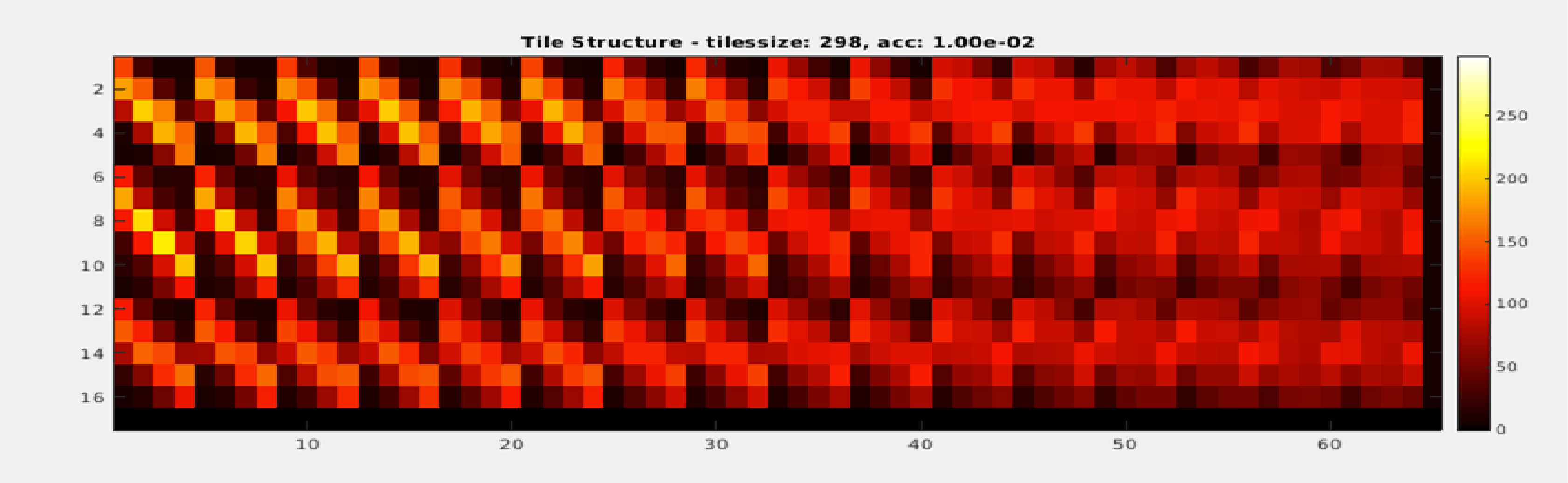
- **Tiles size aligned with system parameters** (¼ of the number of subapertures per WFS -- taking into account circular symmetry properties)
- A vast majority of the tiles have low ranks (i.e., smaller than half of the tile size) => **data sparse**, opportunity for low-rank approximations



Improving performance portability

Ranks analysis: splitting the matrix into tiles and looking at ranks

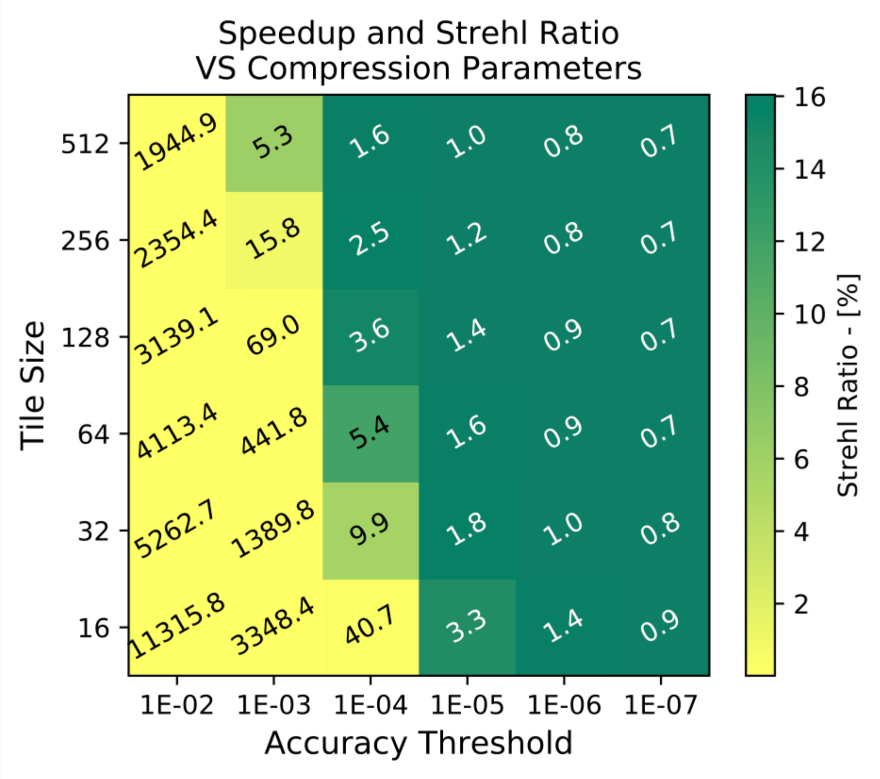
- Tiles size aligned with system parameters
- Mapping the rank / tile across the whole matrix
- Assuming constant tile size, ranks inhomogeneously distributed



Improving performance portability

Accuracy versus tiles size versus speedup

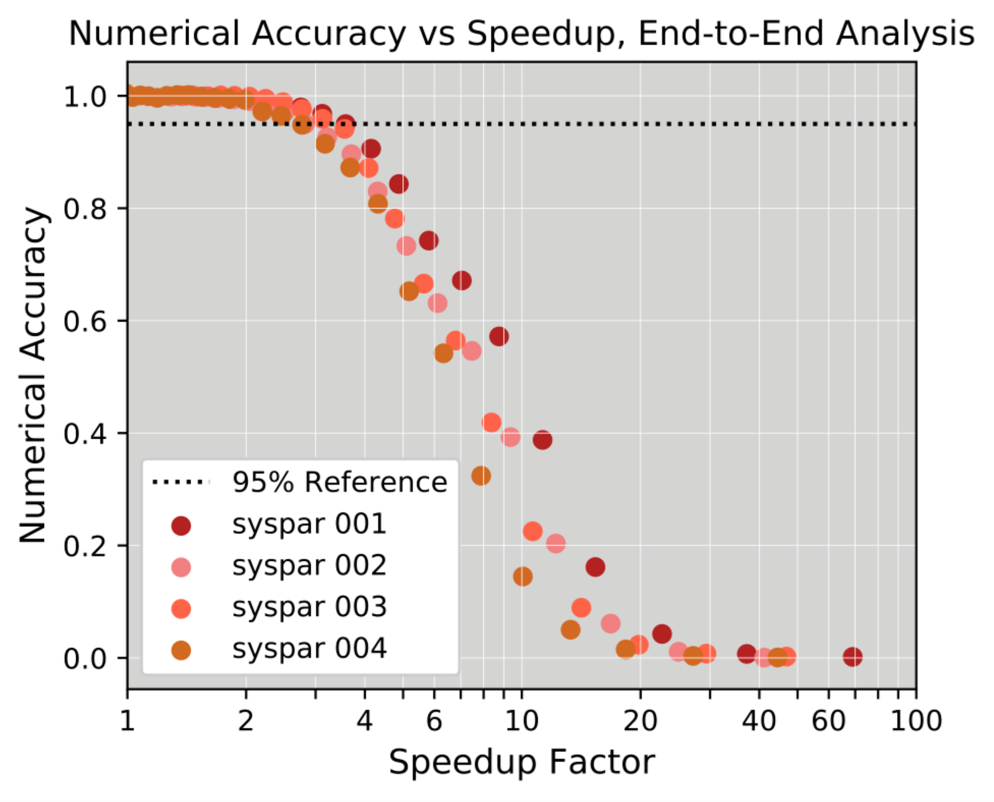
- Exploring compression opportunities (tile sizes & accuracy requirements)



Improving performance portability

Accuracy versus tiles size versus speedup

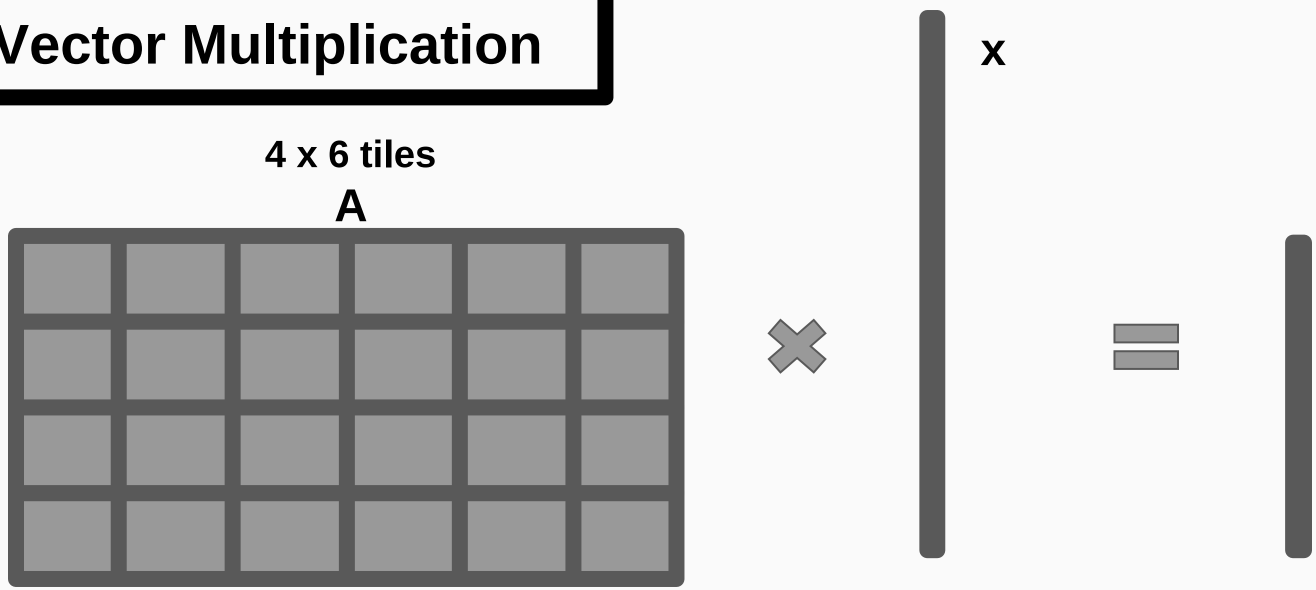
- Exploring compression opportunities (tile sizes & accuracy requirements)
- **x4 speedup with compression leads to acceptable loss in AO performance**



Improving performance portability

How to leverage that ?

Tile Dense Matrix-Vector Multiplication



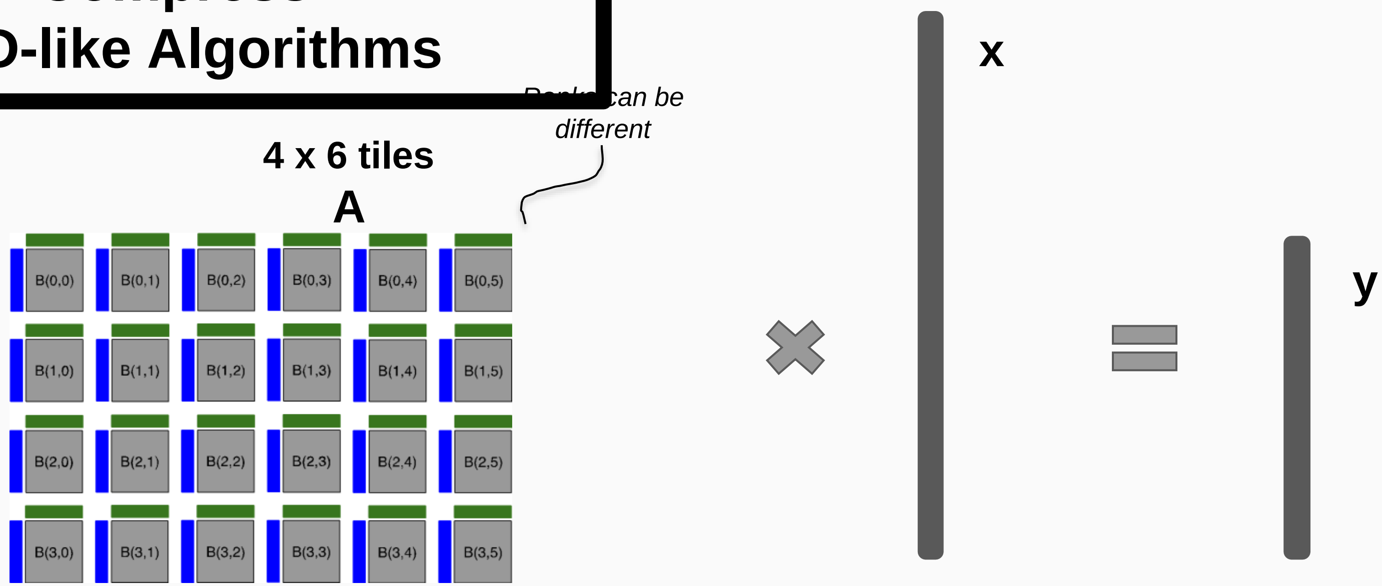
Improving performance portability

Tile Low Rank (TLR) MVM

Compress
SVD-like Algorithms

Ranks can be different

Only once upfront!



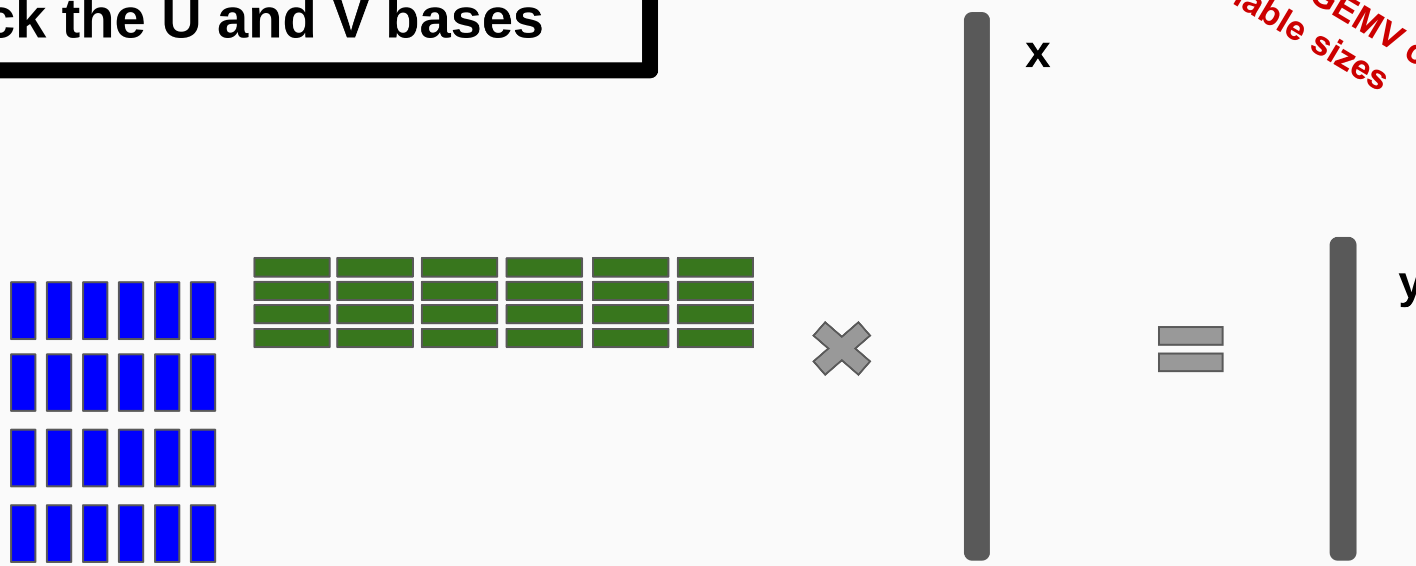
U bases
V bases

Improving performance portability

Tile Low Rank (TLR) MVM

Stack the U and V bases

Rely on batch GEMV calls w/ variable sizes

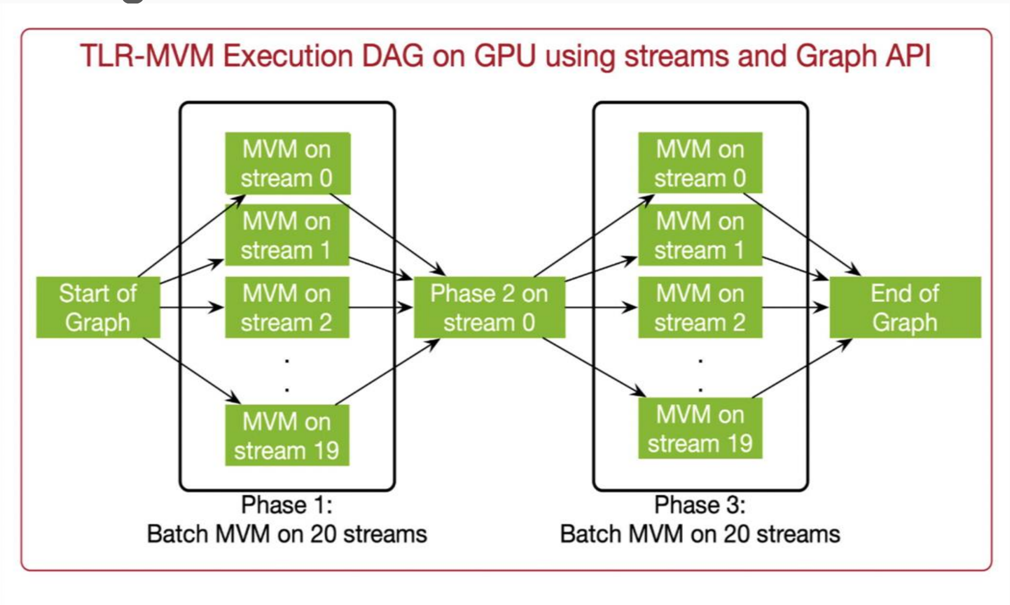


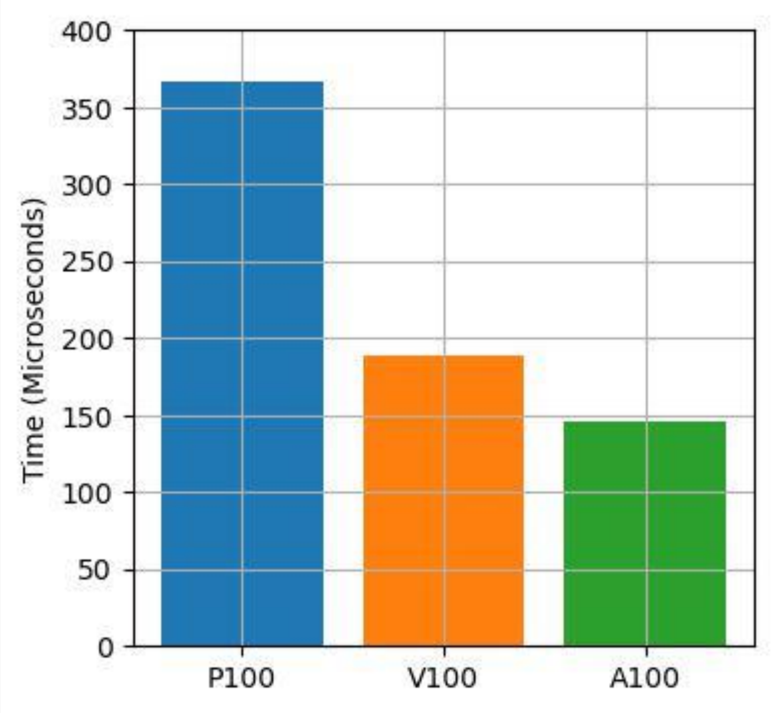
U bases
V bases

Improving performance portability

Tile Low Rank (TLR) MVM: porting on Nvidia GPUs

- Leveraging CUDA streams and Graphs in a single approach
- Assessing performance scalability across several generations
- Sustained speedup > x2: a **single GPU needed to meet performance goals**.





Improving performance portability

Tile Low Rank (TLR) MVM: comparing against hardware landscape

Vendor	Intel	AMD	Fujitsu	NEC	NVIDIA	Graphcore
Family	Cascade Lake	EPYC Milan	Primergy A64FX	SX-Aurora TSUBASA	Ampere GPU	IPU
Model	6248	7713	FX1000	B300-8	A100	Bow
Node(s)/Card(s)	1	1	16	8	1	1
Socket(s)	2	2	4	N/A	N/A	1
Cores	40	128	48	8	6912	1472
GHz	2.5	2.0	2.2	1.6	2.6	1.85
Memory	384GB DDR4	512GB DDR4	32GB HBM	48GB HBM2	40GB HBM2e	3.6GB
Sustained BW	232GB/s	330GB/s	800GB/s	1.5TB/s	1.5TB/s	261TB/s
LLC	27.5MB	512MB	32MB	16MB	40MB	N/A
Sustained BW	1.1TB/s	4TB/s	3.6TB/s	2.1TB/s	4.8TB/s	
Compiler	Intel 19.1.0	GCC 7.5.0	Fujitsu 4.5.0	NEC 3.1.1	NVCC 11.0	POPLAR 2.6
BLAS library	Intel MKL 2020	BLIS 3.0.0	Fujitsu SSL II	NEC NLC 2.1.0	cuBLAS 11.0	N/A
MPI library	OpenMPI 4.0.3	OpenMPI 3.1.2	Fujitsu MPI 4.0.1	NEC MPI 2.13.0	NCCL 2.0	N/A

x86 - ARM - Vector
MPI + OpenMP

GPU
CUDA

Improving performance portability

Tile Low Rank (TLR) MVM: comparing against hardware landscape

Vendor	Intel	AMD	Fujitsu	NEC	NVIDIA	Graphcore
Family	Cascade Lake	EPYC Milan	Primergy A64FX	SX-Aurora TSUBASA	Ampere GPU	IPU
Model	6248	7713	FX1000	B300-8	A100	Bow
Node(s)/Card(s)	1	1	16	8	1	1
Socket(s)	2	2	4	N/A	N/A	1
Cores	40	128	48	8	6912	1472
GHz	2.5	2.0	2.2	1.6	2.6	1.85
Memory Sustained BW	384GB DDR4 232GB/s	512GB DDR4 330GB/s	32GB HBM 800GB/s	48GB HBM2 1.5TB/s	40GB HBM2e 1.5TB/s	3.6GB 261TB/s
LLC Sustained BW	27.5MB 1.1TB/s	512MB 4TB/s	32MB 3.6TB/s	16MB 2.1TB/s	40MB 4.8TB/s	N/A
Compiler	Intel 19.1.0	GCC 7.5.0	Fujitsu 4.5.0	NEC 3.1.1	NVCC 11.0	POPLAR 2.6
BLAS library	Intel MKL 2020	BLIS 3.0.0	Fujitsu SSL II	NEC NLC 2.1.0	cuBLAS 11.0	N/A
MPI library	OpenMPI 4.0.3	OpenMPI 3.1.2	Fujitsu MPI 4.0.1	NEC MPI 2.13.0	NCCL 2.0	N/A

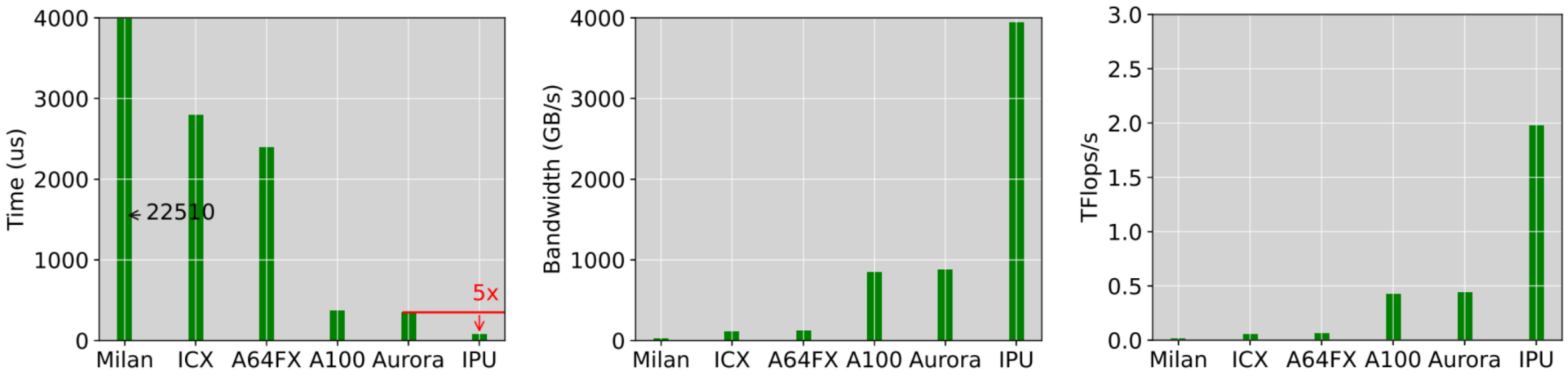
x86 - ARM - Vector
MPI + OpenMP

GPU
CUDA

Improving performance portability

Tile Low Rank (TLR) MVM: steering customized AI hardware from Graphcore

- x5 performance improvement as compared to state-of-the-art high memory bandwidth general purpose processors !
- Opens new opportunities: mixing classical + AI workflows …

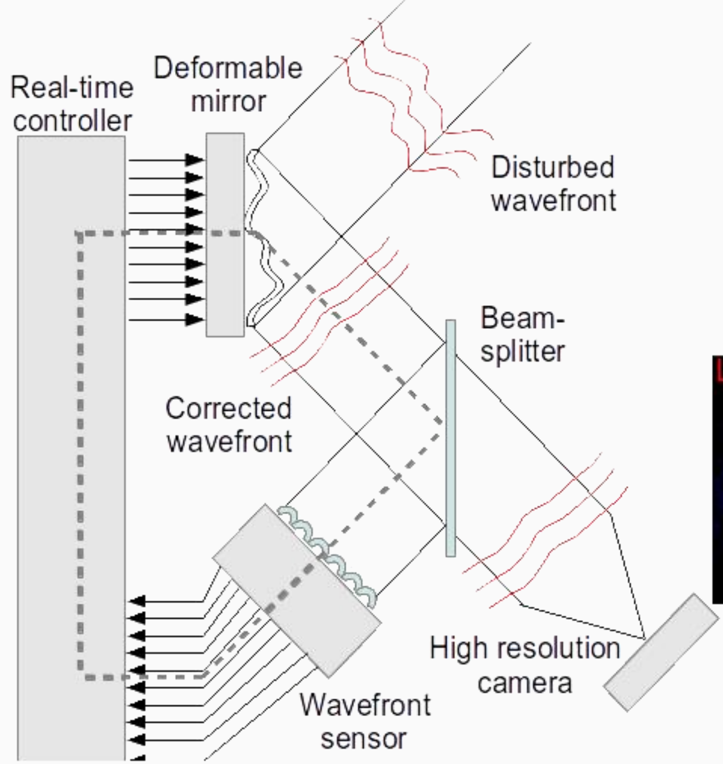


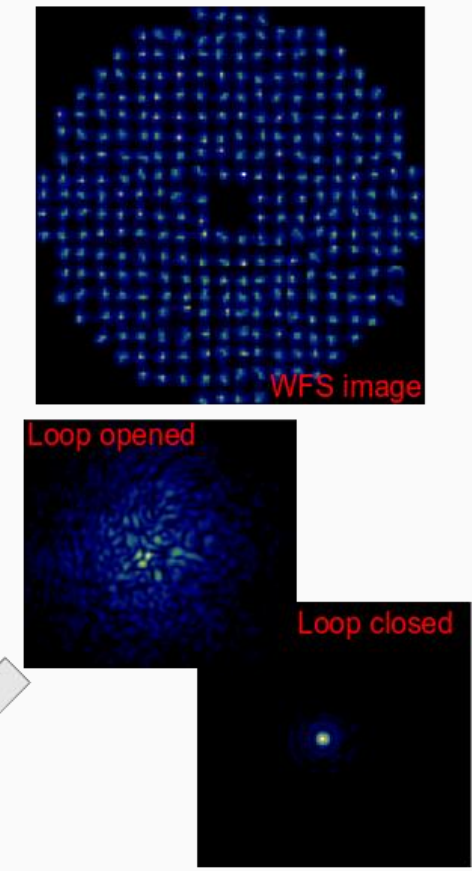
(a) Time-to-solution. (b) Sustained bandwidth. (c) Execution rate.

DESIGNING A NEW BRAIN FOR AO

Complex, multi-physics problem: building a new brain requires multiple flavors of AI mixed with HPC workloads

- **Sensors data:** mitigate noise, improve linearity, merge multiple sensors
- **Deformable mirrors:** improve resolution
- **Pipeline latency:** enable predictive control
- **Variable conditions:** self-adapting controller
- **Stable time-to-solution** real-time inference, deterministic time-to-solution

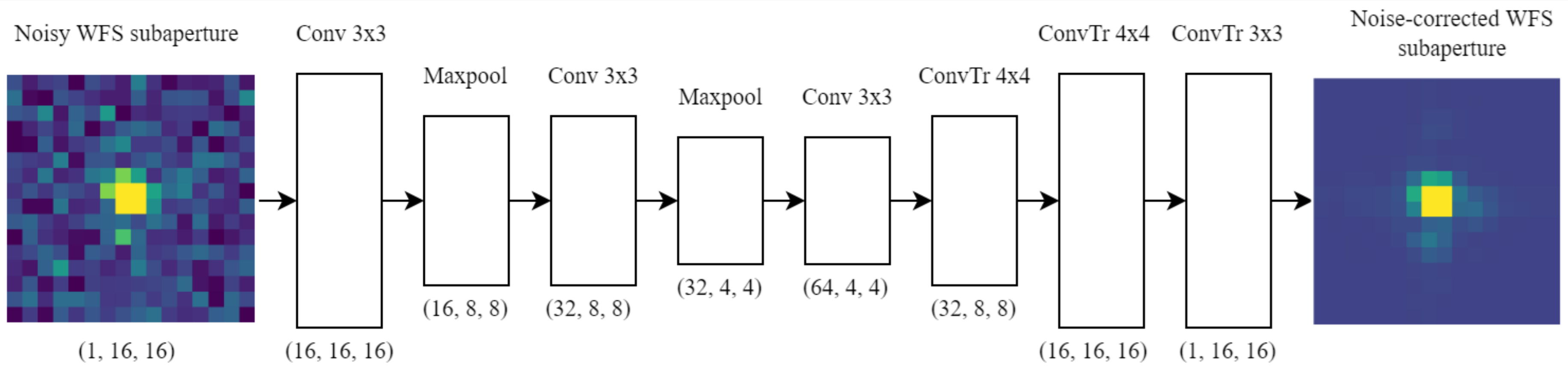




WFS IMAGE DENOISING

Denoising with autoencoders: supervised learning

- Autoencoder: Supervised learning with a sample of noisy frames as input and same frames without noise (obtained with simulator)

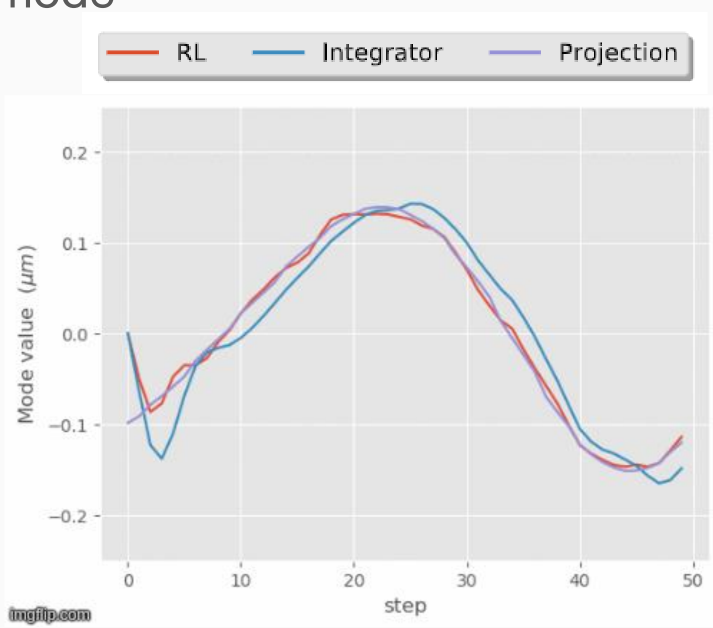


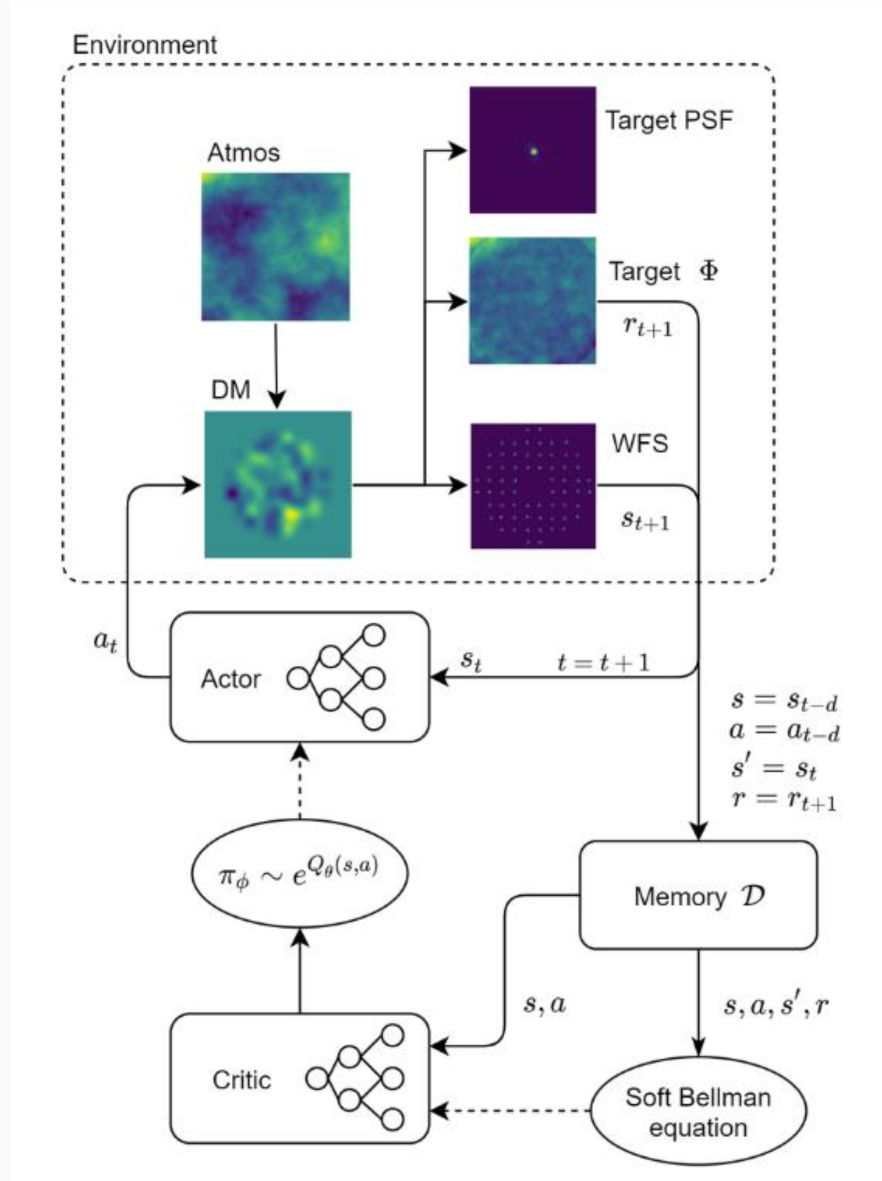
- Could be trained on bench using calibration source tuned to the right brightness
- “Lightweigth” network, could be loaded on the frame grabber itself (looking into FPGA implementation as well as DPU)

AO CONTROL

An example of Reinforcement Learning: predictive control

- Soft Actor-Critic, model free
- Need to define a reward
  - Ideal case: access to phase variance
  - Works with slopes variance
- Exemple time series of commands to a mode





Adaptable, powerful … and Future-Proof

Optimized HPC workflows workflows

- **Performance portability**: leveraging OpenMP + graphs (**see C. Cetre's talk**)
- **Standardized graph-based representation**: using open C++ standards and more (**see J. Bernard's talk**)
- **Leveraging mixed precision**: optimizing Learn & Apply in SRTC for high cadence turbulence profiling (**see N. Doucet's talk**)
- **Towards multi-100 Gb/s data ingestion**: using standard libraries & toolkits (**see J. Plante's talk**)

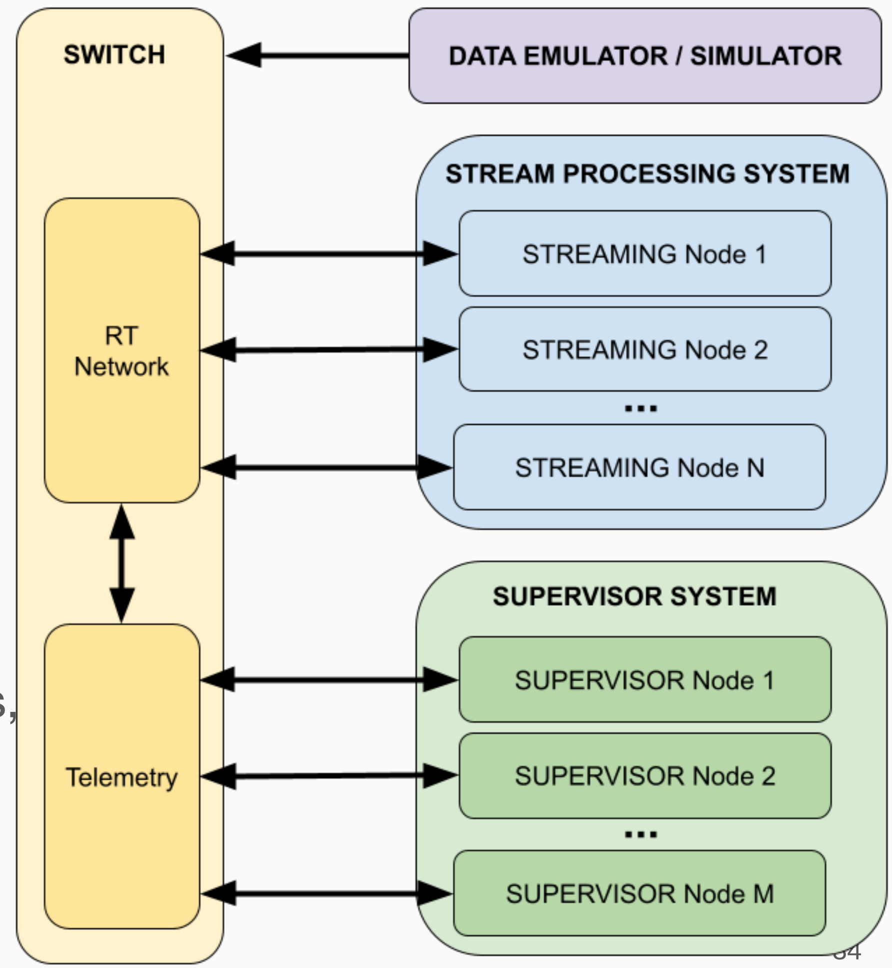
AI integration

- Dual-stage XAO control with generative AI and reinforcement learning: **see B. Pou's talk**
- Super-resolution + PSF reconstruction: **see J. Smith's talk**

SUPPORTING INITIATIVES

STREAMS: a continuous integration platform

- ~1M€ hardware budget
- built around x**10 Tb/s backbone**
- Significant donations from vendors (NVidia, Graphcore)
- Strong contributions from industry partners (Thales, REFLEX CES)
- Collaboration with IDRIS (host), GENCI and other partners
- Additional partnerships being discussed ...
- Will be **testing many technologies**: incl. A100x DPUs, H100 & Mi250 GPUs, Genoa CPUs, Graphcore's IPUs and more



Supporting initiatives

PEPR ORIGINES: we are hiring !

- New AI methodologies and high bandwidth data transport for XAO @ ELT scale
- Several positions opened (PhDs, post-docs, research engineers)
- Please talk with F. Ferreira if interested

FRANCE 2030
PROGRAMME DE RECHERCHE
ORIGINES

That’s it for today !