

# PERFORMANCE ANALYSIS TOOLS @ JSC

## HPC SUPPORT CORNER

NOV 26, 2025 | MICHAEL KNOBLOCH | M.KNOBLOCH@FZ-JUELICH.DE

# OUTLINE

Make it work,  
make it right,  
make it fast.

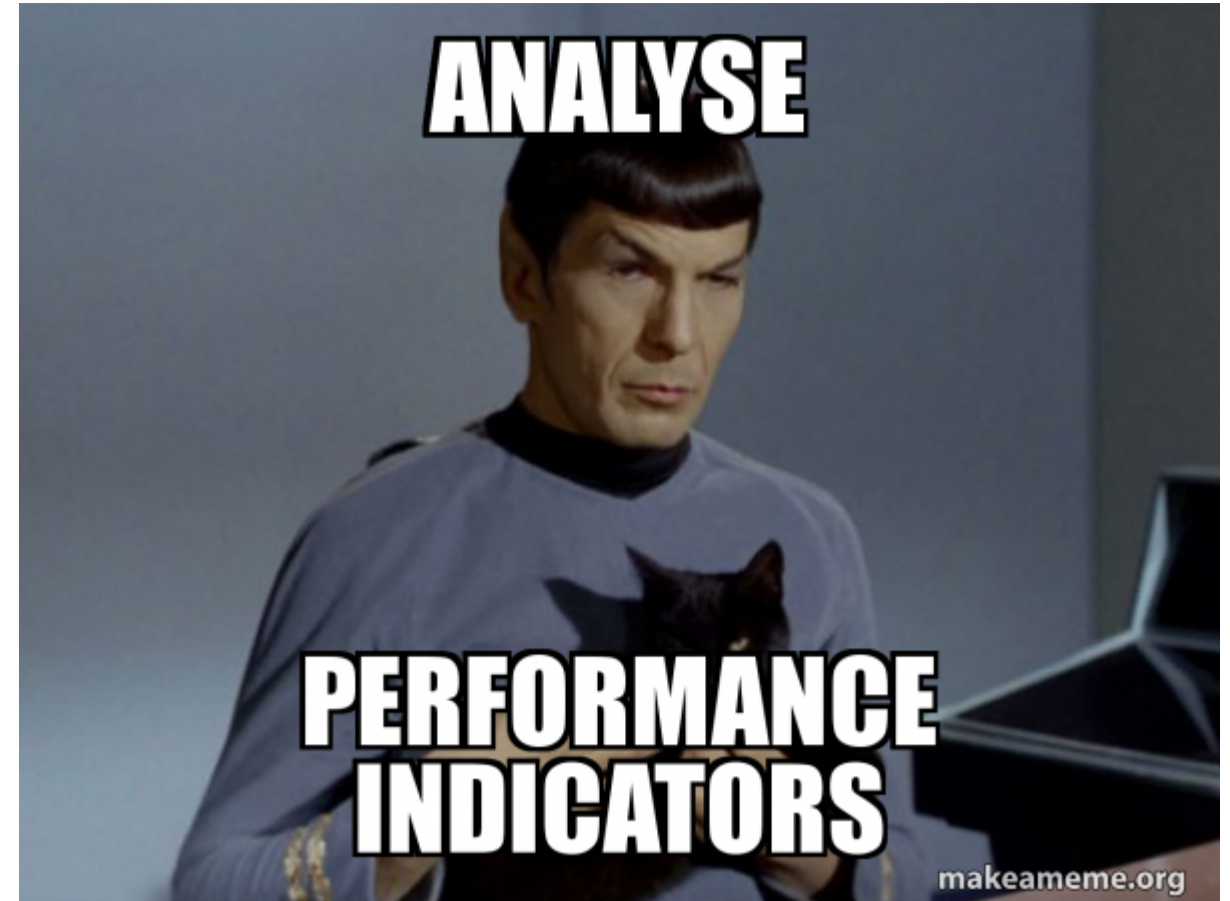
*Kent Beck*

**Debugger and Correctness  
Tools**

**Performance Analysis Tools**

# PERFORMANCE TOOLS (**STATUS: NOV 2025**)

- Score-P
- Scalasca
- Vampir[Server]
- Linaro Forge
  - Performance Reports
  - MAP
- Intel Tools
  - VTune Amplifier XE
  - Intel Advisor
- NVIDIA Tools
  - Nsight Systems
  - Nsight Compute
- Darshan
- ...



# PERFORMANCE ANALYSIS RECOMMENDATIONS

- Measure and analyze at the desired scale (once you have a reasonable measurement setup)
- Get performance overview with Performance Reports (or Intel APS)
  - CPU Issues:
    - Use MAP, Vtune (on Intel nodes), or uProf (on AMD nodes)
    - Use perf / LIKWID / PAPI
  - MPI Issues: Use Scalasca/Vampir
  - GPU Issues: Use NVIDIA tools
  - I/O Issues: Use DARSHAN
- OR: Do it all with Score-P/Scalasca/Vampir

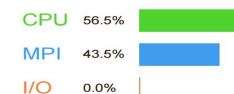
# LINARO PERFORMANCE REPORTS



- Single page report provides quick overview of performance issues
- Works on unmodified, optimized executables
- Shows CPU, memory, network and I/O utilization
- Supports MPI, multi-threading and accelerators
- Saves data in HTML, CVS or text form
- <https://www.linaroforge.com/linaroPerformanceReports>
- Note: License limited to 128 processes (with unlimited number of threads)

Summary: cp2k.popt is **CPU-bound** in this configuration

The total wallclock time was spent as follows:



Time spent running application code. High values are usually good. This is **average**; check the CPU performance section for optimization advice.

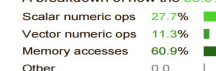
Time spent in MPI calls. High values are usually bad. This is **average**; check the MPI breakdown for advice on reducing it.

Time spent in filesystem I/O. High values are usually bad. This is **negligible**; there's no need to investigate I/O performance.

This application run was **CPU-bound**. A breakdown of this time and advice for investigating further is in the **CPU** section below.

## CPU

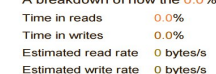
A breakdown of how the **56.5%** total CPU time was spent:



The per-core performance is memory-bound. Use a profiler to identify time-consuming loops and check their cache performance. Little time is spent in **vectorized instructions**. Check the compiler's vectorization advice to see why key loops could not be vectorized.

## I/O

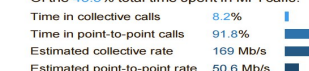
A breakdown of how the **0.0%** total I/O time was spent:



No time is spent in **I/O operations**. There's nothing to optimize here!

## MPI

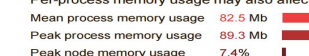
Of the **43.5%** total time spent in MPI calls:



The point-to-point transfer rate is low. This can be caused by inefficient message sizes, such as many small messages, or by imbalanced workloads causing processes to wait. Use an MPI profiler to identify the problematic calls and ranks.

## Memory

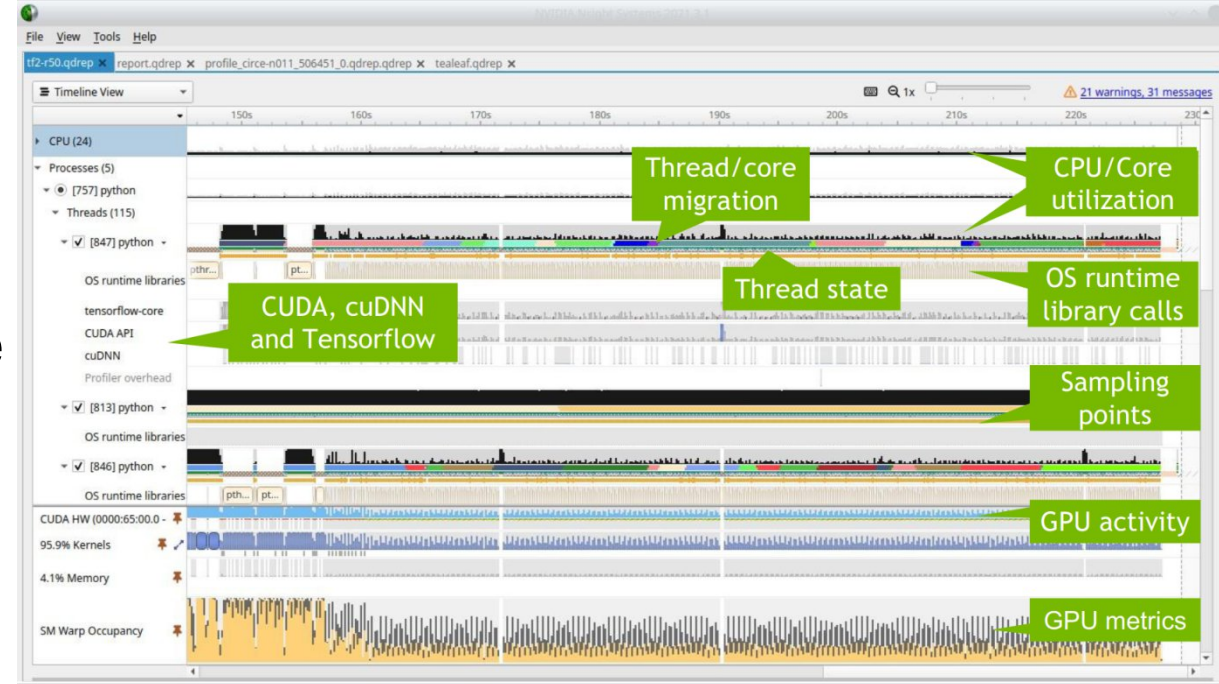
Per-process memory usage may also affect scaling:



The peak node memory usage is low. You may be able to reduce the total number of CPU hours used by running with fewer MPI processes and more data on each process.

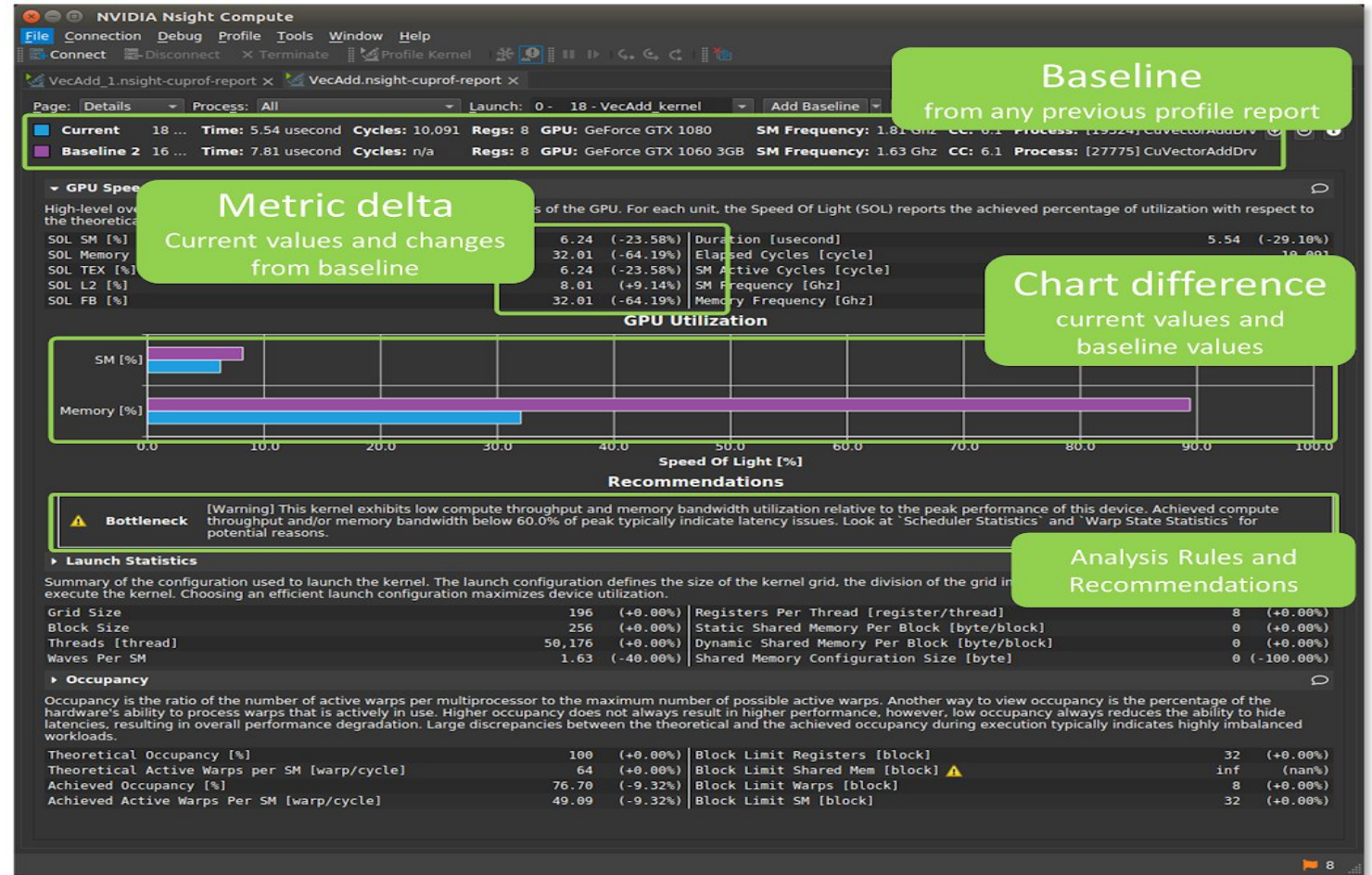
# NSIGHT SYSTEM

- System-wide application tuning
- Locate optimization opportunities
  - Visualize millions of events on a timeline
  - See gaps of unused CPU and GPU time
- Balance workloads across multiple CPUs and GPUs
  - CPU utilization and thread state
  - GPU streams, kernels, memory transfers, etc.
- Multi-platform support
  - Linux, Windows and Mac OS X (host-only)
  - x86-64, Power9, ARM server, Tegra (Linux & QNX)



# NSIGHT COMPUTE

- Interactive kernel profiler
- Detailed performance metrics
- Guided analysis
- Baseline feature to compare versions
- Customizable and data-driven UI
- Supports analysis scripts for post-processing results
- CLI and GUI



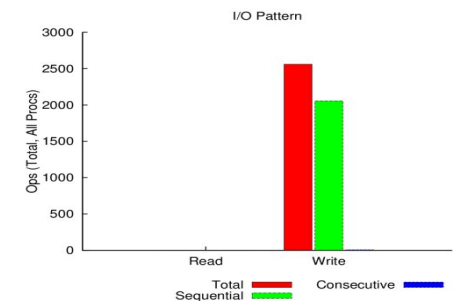
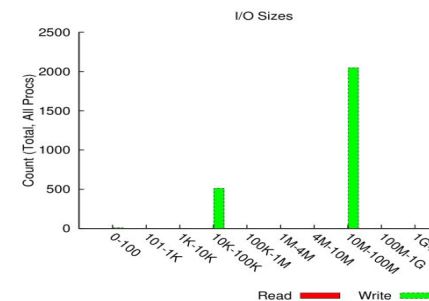
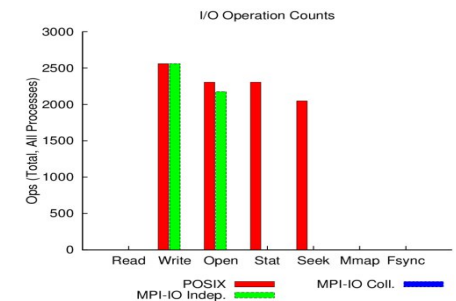
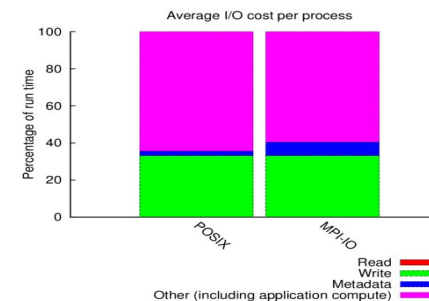
# DARSHAN

- I/O characterization tool logging parallel application file access
- Summary report provides quick overview of performance issues
- Works on unmodified, optimized executables
- Shows counts of file access operations, times for key operations, histograms of accesses, etc.

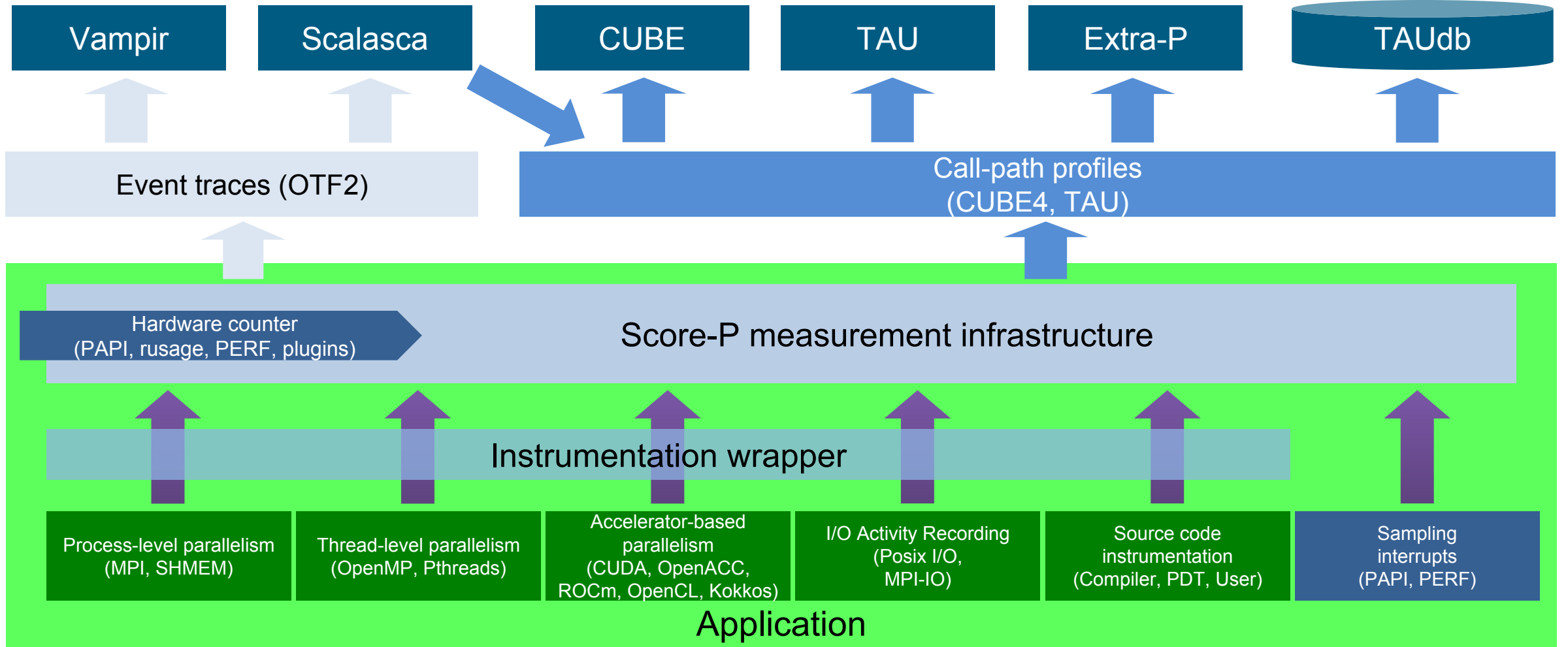
- Supports POSIX, MPI-IO, HDF5, PnetCDF, ...
- Binary log file written at exit post-processed into PDF

- <http://www.mcs.anl.gov/research/projects/darshan/>
- Open Source: installed on many HPC systems

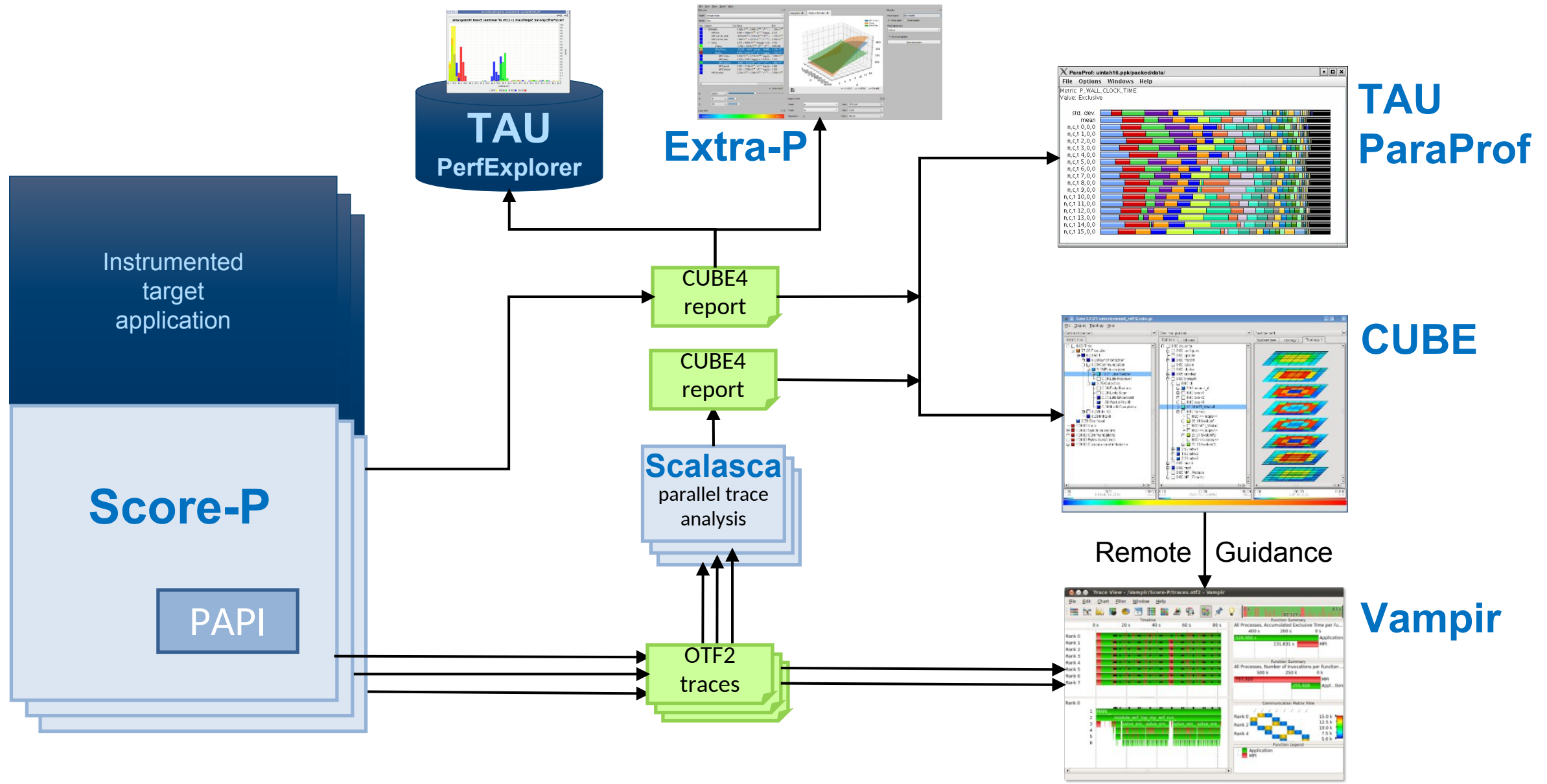
jobid: | uid: | nprocs: 4096 | runtime: 175 seconds



# Score-P ARCHITECTURE

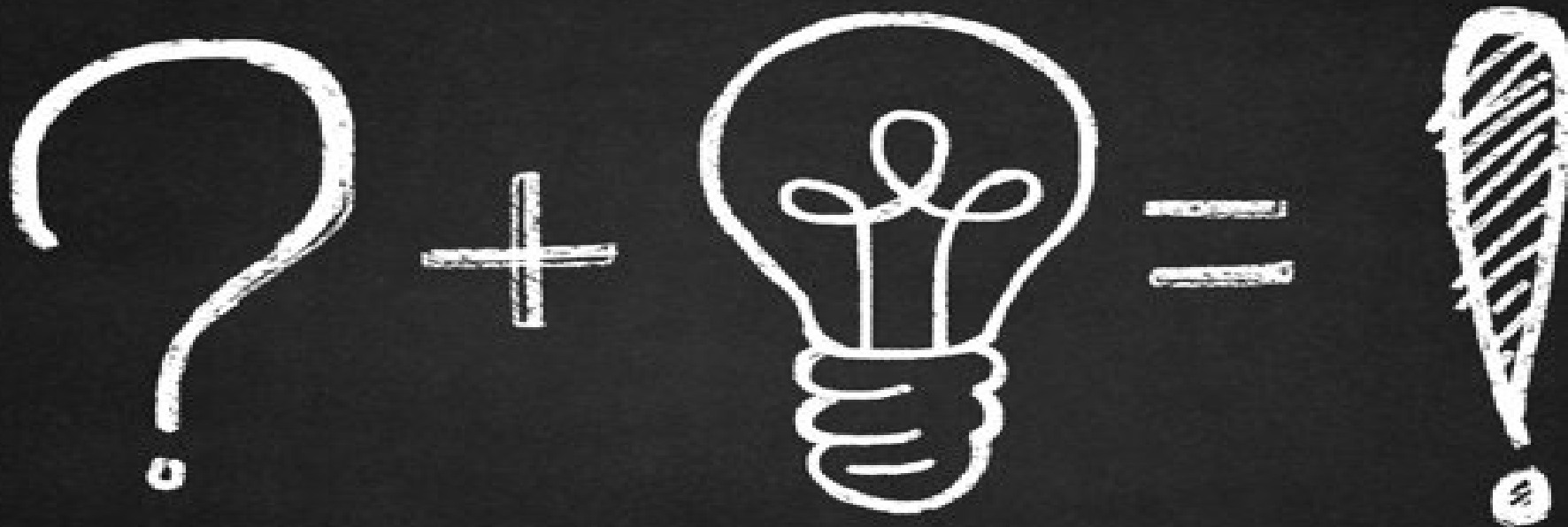


# Score-P TOOL ECOSYSTEM



# NEED HELP?

- Talk to the experts
  - Use local 1<sup>st</sup>-level support, e.g. SC support, SimLab
  - Use mailing lists
  - JSC/NVIDIA Application Lab
  - ATML Parallel Performance
    - VI-HPS Tuning Workshops
    - POP CoE
  - ATML Application Optimization and User Service Tools
    - EPICURE
- Successful performance engineering often is a collaborative effort



# QUESTIONS