Calculating POP Metrics with Scalasca

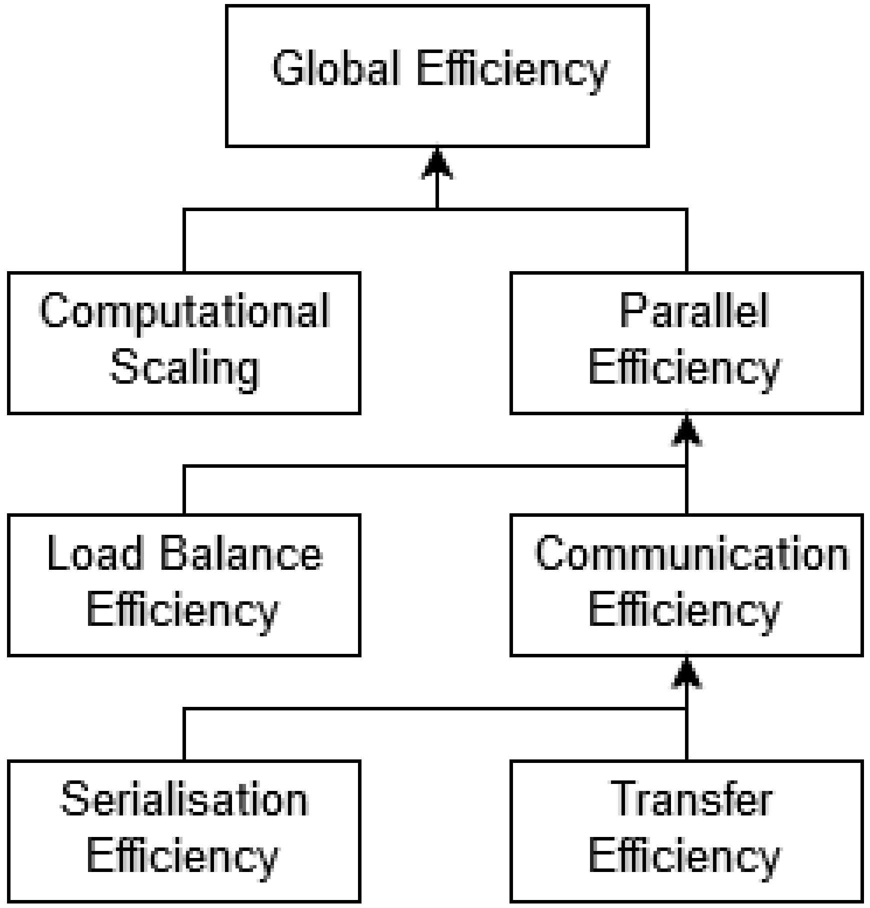
Michael Knobloch
Jülich Supercomputing Centre

POP Metrics

https://pop-coe.eu/further-information/learning-material

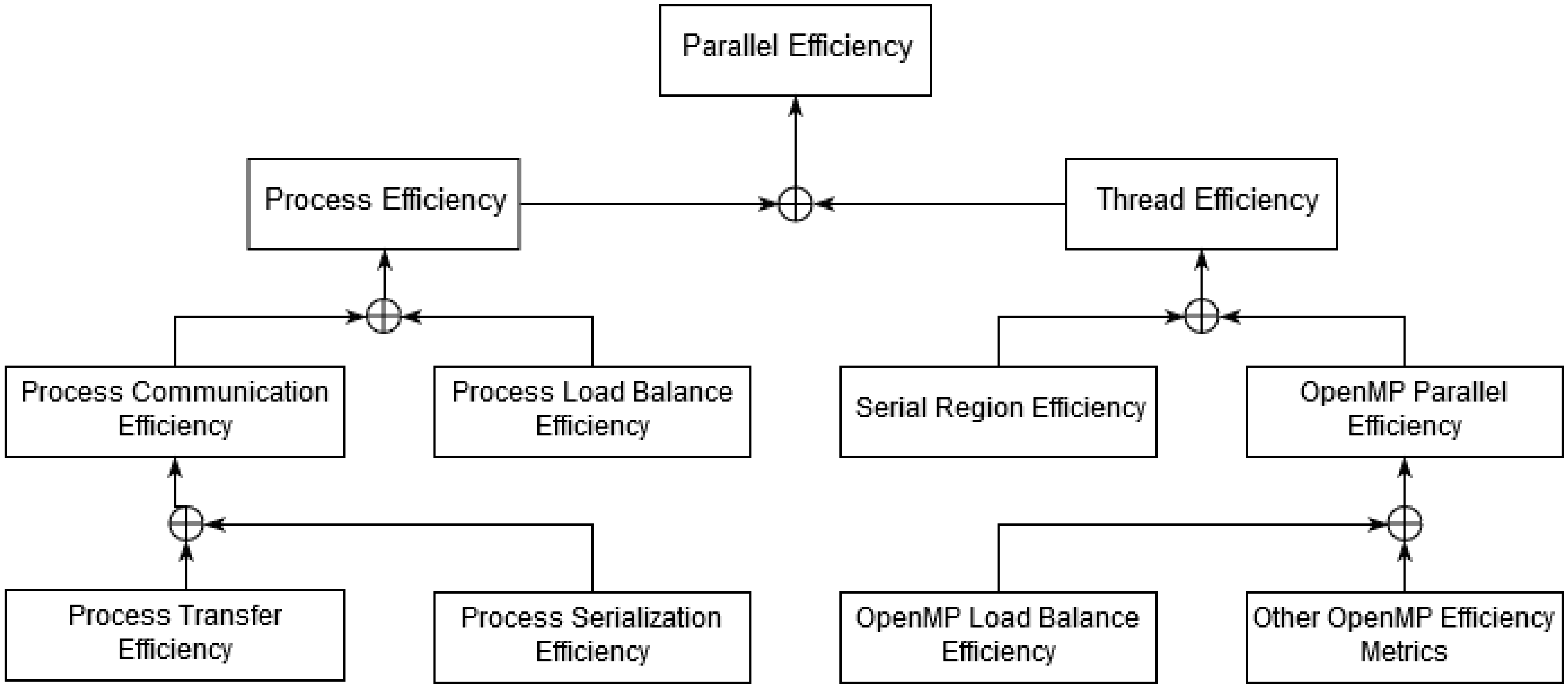
- **Original (POP1) Metrics**
  - Article explaining the POP Standard Metrics for Parallel Performance Analysis
  - Presentation summarizing the POP Standard Metrics for Parallel Performance Analysis
- **New (POP2) Hybrid Metrics**
  - Introduction explaining the POP2 Standard Metrics for Performance Analysis of Hybrid Parallel Applications
  - Cheat sheet for Additive Hybrid Metrics
  - Cheat sheet for Multiplicative Hybrid Metrics
  - In-depth explanation of the Additive Hybrid Metrics
  - Webinar Identifying Performance Bottlenecks in Hybrid MPI + OpenMP Software

Original POP Metrics

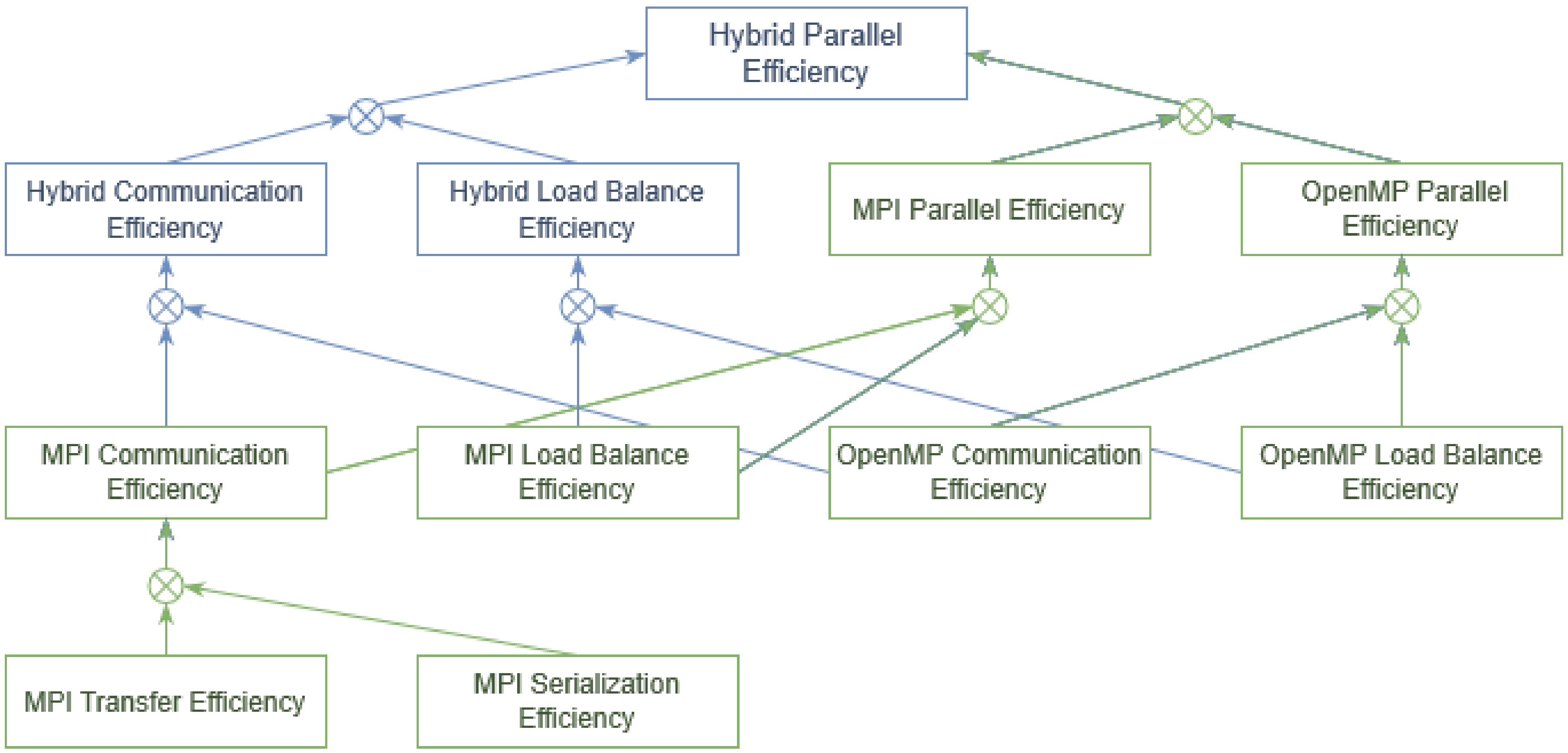


- **Parallel Efficiency (PE):** reveals the inefficiency in splitting computation over processes and communicating data
  - **PE = LB * CE**
- **Load Balance (LB):** ratio of average useful computation time (across all processes) and maximum comp. time
  - LB = avg. useful comp. time / max. comp. time
- **Communication Efficiency (CE):** ratio of maximum comp. time (across all processes) and total runtime
  - CE = SerE * TE = max. comp. time / tot. runtime
- **Serialisation Efficiency (SerE):** describes loss of efficiency due to dependencies between processes causing other processes to wait
  - SerE = max comp. time on ideal network / total runtime on ideal network
- **Transfer Efficiency (TE):** represents inefficiencies due to time in data transfer
  - TE = tot. Runtime on ideal network / tot runtime on real network

Additive Hybrid POP Metrics



Multiplicative Hybrid POP Metrics



POP Metrics + Scalasca

1. Instrument application and setup measurement parameters (e.g. filtering)
   - `scorep <comp+link+cmds>`
   - `scan <exec+cmd> …`
2. For parallel efficiency: perform trace measurement and analysis
3. For computational scaling: perform profile measurement with suitable HW counters
   - `scan –P pop <exec+cmd>`
4. Merge profile and trace measurement
5. Post-process measurement
6. Analyze POP metrics with Cube Advisor
   - `square <measurement+archive>`

Requires
- Scalasca >= V2.6
- Cube >= V4.6

!

Measurement Example

- Measurement of simple Jacobi solver
  - Solves Poisson equation on rectangular grid assuming
    - Uniform discretization in each direction
    - Dirichlect boundary conditions

- Available in multiple variants (Shipped with Score-P)
  - C, C++ or Fortran source code
  - MPI, OpenMP, or hybrid (MPI+OpenMP)

POP Preset Measurement

openSUSE-Leap-15-1

```
zam310:~/jacobi/hybrid/C [1] scan -P pop mpiexec -np 2 ./jacobi
path:>/opt/local/Scalasca-2.6/share/scalasca/presets/pop.presets
S=C=A=N: RUN: 0 REPETITION: 0 Archive=./scorep_jacobi_2x2_preset_pop_c2/scorep_jacobi_2x2_c1
S=C=A=N: RUN:  Scalasca tracing run
S=C=A=N: Scalasca 2.6 trace collection and analysis
S=C=A=N: ./scorep_jacobi_2x2_preset_pop_c2/scorep_jacobi_2x2_c1 experiment archive
S=C=A=N: Tue May 25 16:35:02 2021: Collect start
/opt/local/easybuild-4.1.1/software/OpenMPI/3.1.4-GCC-system-2.31/bin/mpiexec -np 2 ./jacobi
Jacobi 2 MPI-3.1#1 process(es) with 2 OpenMP-201511 thread(s)/process

-> matrix size: 2000x2000
-> alpha: 0.800000
-> relax: 1.000000
-> tolerance: 0.000000
-> iterations: 100

 Number of iterations : 100
 Residual             : 5.955111e-10
 Solution Error       : 0.000266483315
 Elapsed Time         : 3.1198404
 MFlops               : 1663.420090
S=C=A=N: Tue May 25 16:35:06 2021: Collect done (status=0) 4s
S=C=A=N: Tue May 25 16:35:06 2021: Analyze start
/opt/local/easybuild-4.1.1/software/OpenMPI/3.1.4-GCC-system-2.31/bin/mpiexec -np 2 /opt/local/Scalasca-2.6/bin/scout.hy
b ./scorep_jacobi_2x2_preset_pop_c2/scorep_jacobi_2x2_c1/traces.otf2
SCOUT   (Scalasca 2.6)
Copyright (c) 1998-2021 Forschungszentrum Juelich GmbH
Copyright (c) 2014-2021 RWTH Aachen University
Copyright (c) 2009-2014 German Research School for Simulation Sciences GmbH

Analyzing experiment archive ./scorep_jacobi_2x2_preset_pop_c2/scorep_jacobi_2x2_c1/traces.otf2

Opening experiment archive ... done (0.000s).
Reading definition data    ... done (0.001s).
Reading event trace data   ... done (0.004s).
```

Notes
- -P pop selects POP metrics measurement
- Automatically executes necessary trace and profile measurements

POP Preset Measurement

openSUSE-Leap-15-1

```
Max. memory usage         : 0.000MB

Total processing time     : 0.119s
S=C=A=N: Tue May 25 16:35:07 2021: Analyze done (status=0) 1s
Warning: 0.172MB of analyzed trace data retained in ./scorep_jacobi_2x2_preset_pop_c2/scorep_jacobi_2x2_c1/traces!
S=C=A=N: ./scorep_jacobi_2x2_preset_pop_c2/scorep_jacobi_2x2_c1 complete.

S=C=A=N: RUN: 1 REPETITION: 0 Archive=./scorep_jacobi_2x2_preset_pop_c2/scorep_jacobi_2x2_c2
S=C=A=N: RUN:  Profiling run with PAPI counters
S=C=A=N: Scalasca 2.6 runtime summarization
S=C=A=N: ./scorep_jacobi_2x2_preset_pop_c2/scorep_jacobi_2x2_c2 experiment archive
S=C=A=N: Tue May 25 16:35:07 2021: Collect start
/opt/local/easybuild-4.1.1/software/OpenMPI/3.1.4-GCC-system-2.31/bin/mpiexec -np 2 ./jacobi
Jacobi 2 MPI-3.1#1 process(es) with 2 OpenMP-201511 thread(s)/process

-> matrix size: 2000x2000
-> alpha: 0.800000
-> relax: 1.000000
-> tolerance: 0.000000
-> iterations: 100

 Number of iterations : 100
 Residual             : 5.955111e-10
 Solution Error       : 0.000266483315
 Elapsed Time         : 3.0536247
 MFlops               : 1699.490183
S=C=A=N: Tue May 25 16:35:10 2021: Collect done (status=0) 3s
S=C=A=N: ./scorep_jacobi_2x2_preset_pop_c2/scorep_jacobi_2x2_c2 complete.
zam310:~/jacobi/hybrid/C [2] square -s ./scorep_jacobi_2x2_preset_pop_c2
INFO: Merging aggregated runtime summary and trace analysis reports...
INFO: Post-processing combined summary and trace analysis report (scout+profile.cubex)...
/opt/local/ScoreP-7.0/bin/scorep-score  -r ./scorep_jacobi_2x2_preset_pop_c2/scout+profile.cubex > ./scorep_jacobi_2x2_p
reset_pop_c2/scorep.score
INFO: Score report written to ./scorep_jacobi_2x2_preset_pop_c2/scorep.score
zam310:~/jacobi/hybrid/C [3]
```

Notes

- Square recognizes POP metrics measurement
- Merges the two measurements before post-processing

Analysis Example

- TeaLeaf Reference V1.0

- HPC mini-app developed by the UK Mini-App Consortium
  - Solves the linear 2D heat conduction equation on a spatially decomposed regular grid using a 5 point stencil with implicit solvers
  - https://github.com/UK-MAC/TeaLeaf_ref/archive/v1.0.tar.gz

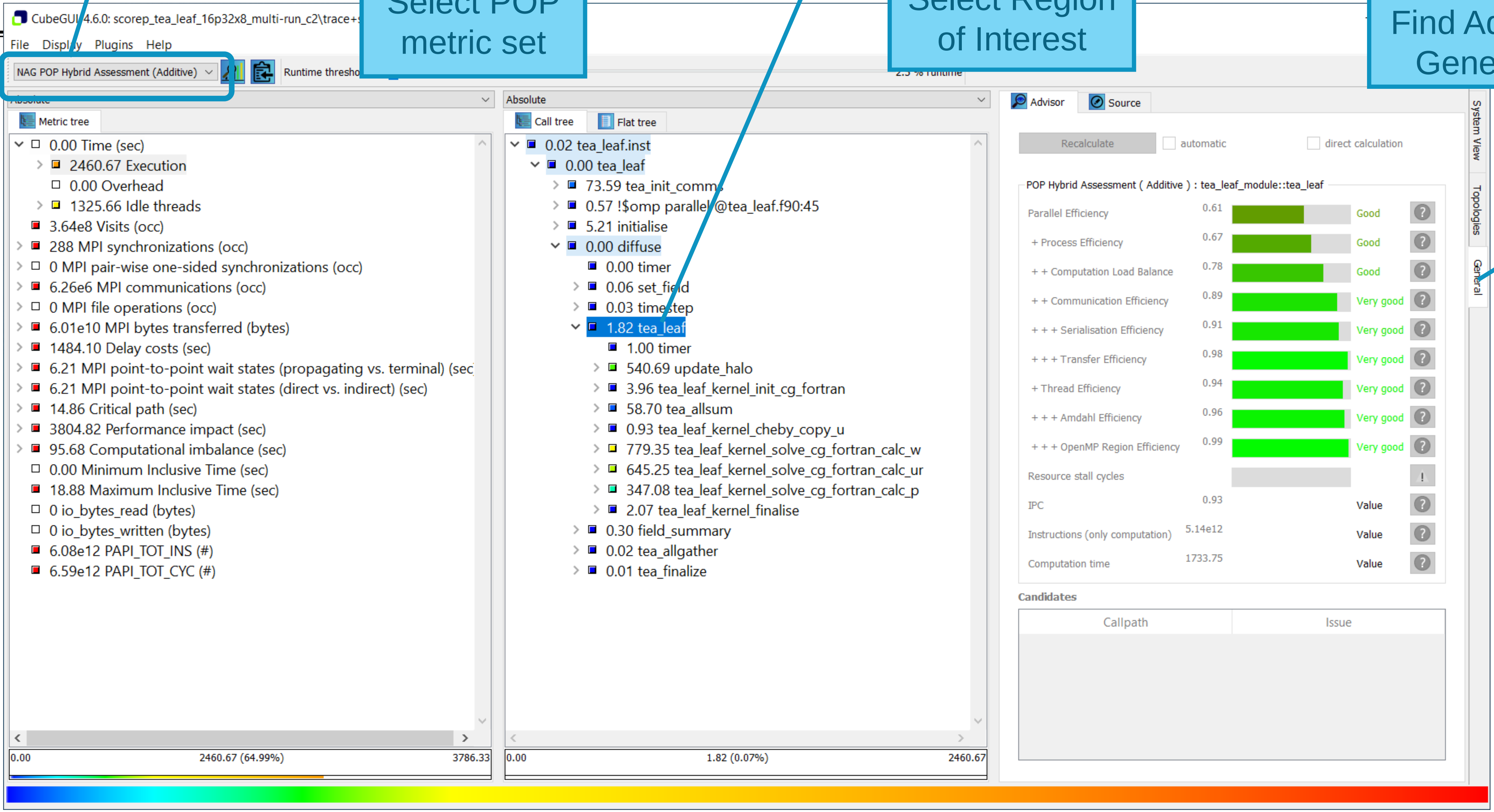
- Measurements performed on Jusuf cluster @ JSC
  - Run configuration
    - 32 MPI ranks with 8 OpenMP threads each
    - Distributed across 2 compute nodes (16 ranks per node)
    - Test problem “5”: 4000 × 4000 cells, CG solver

POP Metrics + CUBE Advisor

Select POP metric set

Select Region of Interest

Find Advisor on General Tab



Hands-on:
NPB-MZ-MPI / BT

cube
scalasca