

Calculating POP Metrics with Scalasca

Michael Knobloch
Jülich Supercomputing Centre



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

POP Metrics

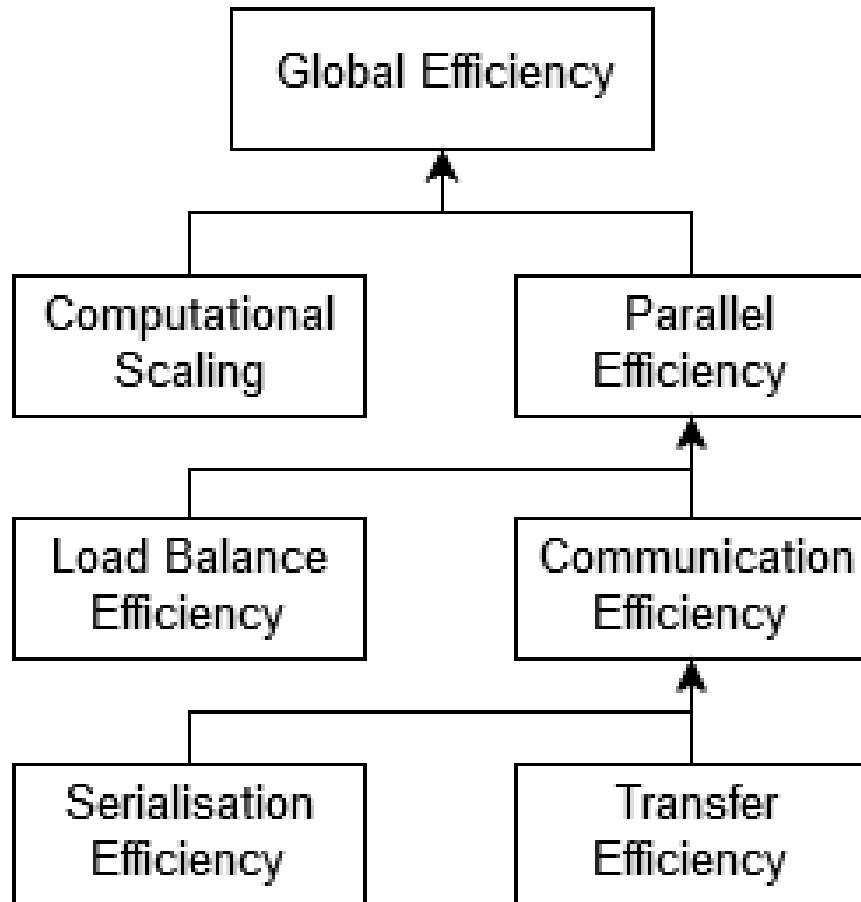
- **Original (POP1) Metrics**

- Article explaining the POP Standard Metrics for Parallel Performance Analysis
- Presentation summarizing the POP Standard Metrics for Parallel Performance Analysis

- **New (POP2) Hybrid Metrics**

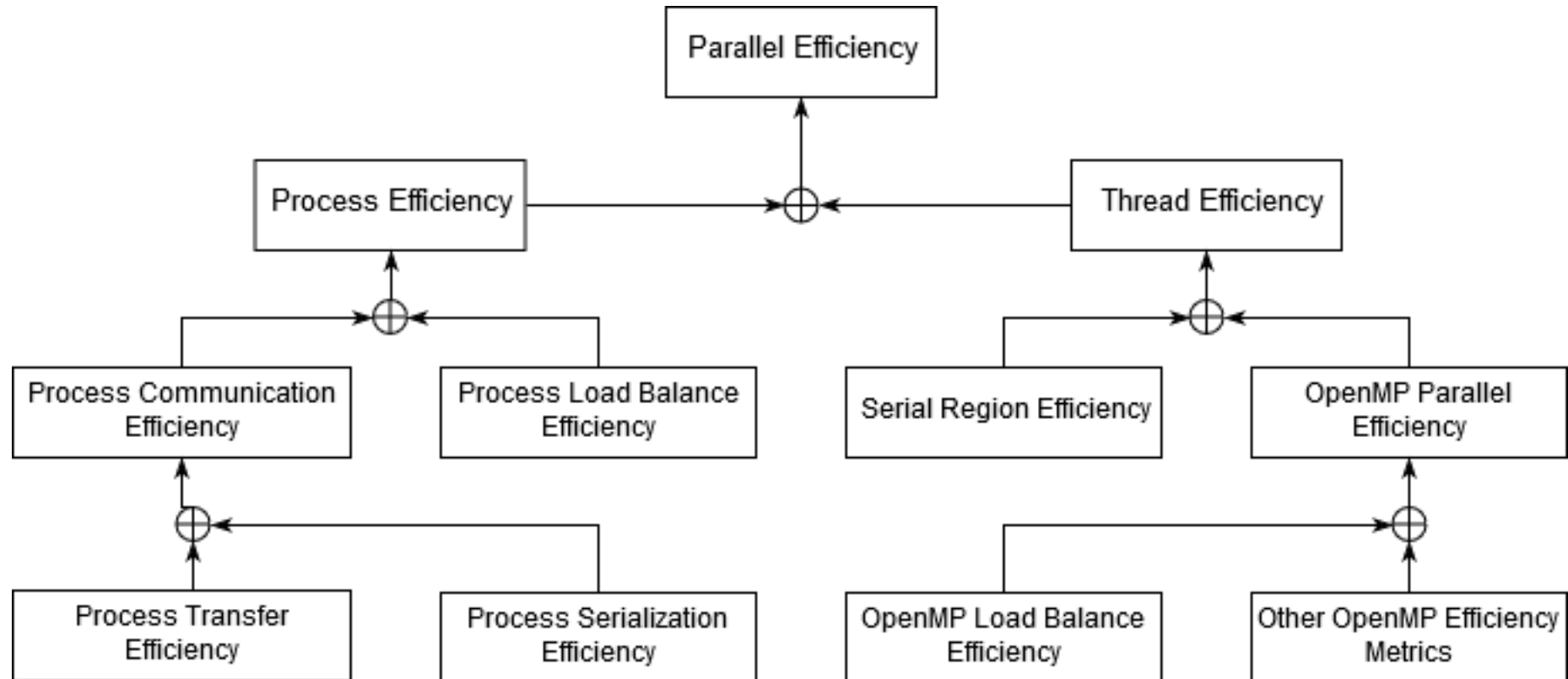
- ➡ • Introduction explaining the POP2 Standard Metrics for Performance Analysis of Hybrid Parallel Applications
- Cheat sheet for Additive Hybrid Metrics
- Cheat sheet for Multiplicative Hybrid Metrics
- In-depth explanation of the Additive Hybrid Metrics
- ➡ • Webinar Identifying Performance Bottlenecks in Hybrid MPI + OpenMP Software

Original POP Metrics

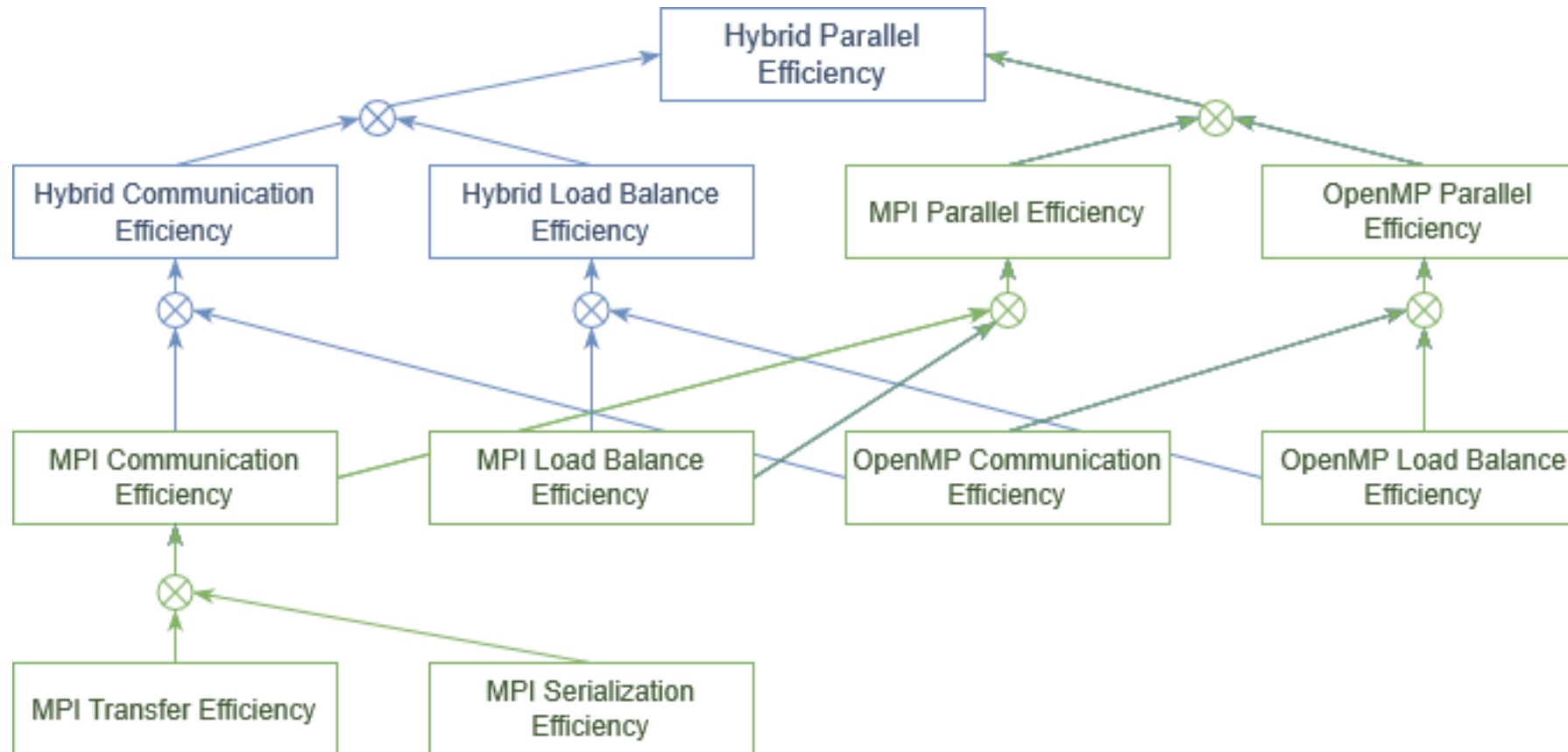


- **Parallel Efficiency (PE):** reveals the inefficiency in splitting computation over processes and communicating data
 - $PE = LB * CE$
- **Load Balance (LB):** ratio of average useful computation time (across all processes) and maximum comp. time
 - $LB = \text{avg. useful comp. time} / \text{max.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

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 \geq V2.6
- Cube \geq V4.6



Measurement Example

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

```
zam310:~/jacobi/hybrid/C [1] scan -P pop mpiexec -np 2 ./jacobi
path:~/opt/local/Scalasca-2.6/share/scalasca/presets/pop.preset<
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_preset_pop_c2/scorep.score
INFO: Score report written to ./scorep_jacobi_2x2_preset_pop_c2/scorep.score
zam310:~/jacobi/hybrid/C [3]
```

Notes

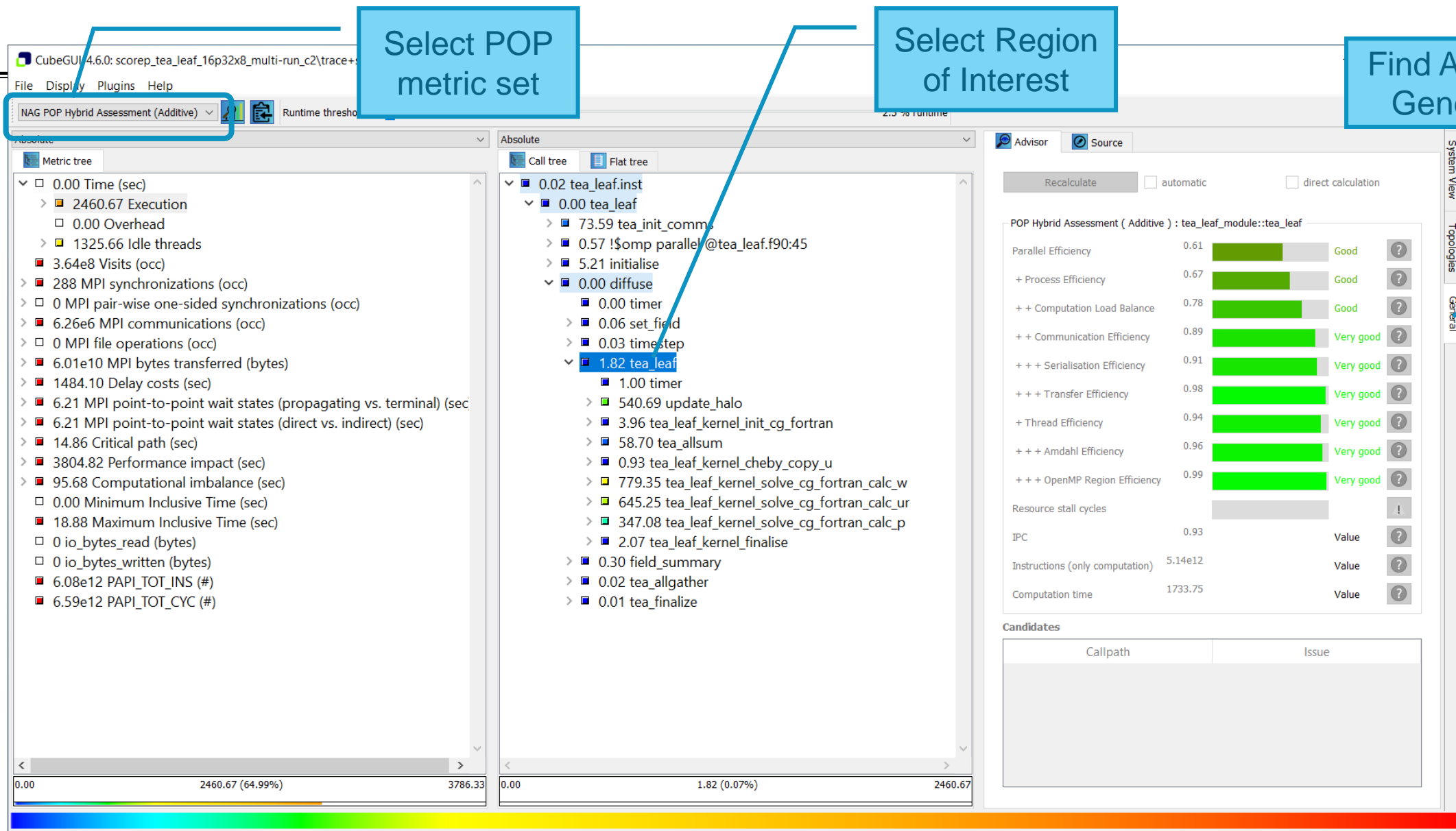
- Square recognizes POP metrics measurement
- Merges the two measurements before post-processing

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

CubeGUI 4.6.0: scorep_tea_leaf_16p32x8_multi-run_c2\trace+

File Display Plugins Help

NAG POP Hybrid Assessment (Additive) Runtime threshold 2.5 % runtime

Metric tree

- 0.00 Time (sec)
 - 2460.67 Execution
 - 0.00 Overhead
 - 1325.66 Idle threads
 - 3.64e8 Visits (occ)
 - 288 MPI synchronizations (occ)
 - 0 MPI pair-wise one-sided synchronizations (occ)
 - 6.26e6 MPI communications (occ)
 - 0 MPI file operations (occ)
 - 6.01e10 MPI bytes transferred (bytes)
 - 1484.10 Delay costs (sec)
 - 6.21 MPI point-to-point wait states (propagating vs. terminal) (sec)
 - 6.21 MPI point-to-point wait states (direct vs. indirect) (sec)
 - 14.86 Critical path (sec)
 - 3804.82 Performance impact (sec)
 - 95.68 Computational imbalance (sec)
 - 0.00 Minimum Inclusive Time (sec)
 - 18.88 Maximum Inclusive Time (sec)
 - 0 io_bytes_read (bytes)
 - 0 io_bytes_written (bytes)
 - 6.08e12 PAPI_TOT_INS (#)
 - 6.59e12 PAPI_TOT_CYC (#)

Absolute

Call tree Flat tree

- 0.02 tea_leaf.inst
 - 0.00 tea_leaf
 - 73.59 tea_init_comms
 - 0.57 !\$omp parallel @tea_leaf.f90:45
 - 5.21 initialise
 - 0.00 diffuse
 - 0.00 timer
 - 0.06 set_field
 - 0.03 timestep
 - 1.82 tea_leaf
 - 1.00 timer
 - 540.69 update_halo
 - 3.96 tea_leaf_kernel_init_cg_fortran
 - 58.70 tea_allsum
 - 0.93 tea_leaf_kernel_cheby_copy_u
 - 779.35 tea_leaf_kernel_solve_cg_fortran_calc_w
 - 645.25 tea_leaf_kernel_solve_cg_fortran_calc_ur
 - 347.08 tea_leaf_kernel_solve_cg_fortran_calc_p
 - 2.07 tea_leaf_kernel_finalise
 - 0.30 field_summary
 - 0.02 tea_allgather
 - 0.01 tea_finalize

Advisor Source

Recalculate automatic direct calculation

POP Hybrid Assessment (Additive) : tea_leaf_module::tea_leaf

Metric	Value	Efficiency	Quality
Parallel Efficiency	0.61	<div style="width: 61%;"></div>	Good
+ Process Efficiency	0.67	<div style="width: 67%;"></div>	Good
++ Computation Load Balance	0.78	<div style="width: 78%;"></div>	Good
+++ Communication Efficiency	0.89	<div style="width: 89%;"></div>	Very good
+++ Serialisation Efficiency	0.91	<div style="width: 91%;"></div>	Very good
+++ Transfer Efficiency	0.98	<div style="width: 98%;"></div>	Very good
+ Thread Efficiency	0.94	<div style="width: 94%;"></div>	Very good
+++ Amdahl Efficiency	0.96	<div style="width: 96%;"></div>	Very good
+++ OpenMP Region Efficiency	0.99	<div style="width: 99%;"></div>	Very good
Resource stall cycles			!
IPC	0.93		Value
Instructions (only computation)	5.14e12		Value
Computation time	1733.75		Value

Candidates

Callpath	Issue

0.00 2460.67 (64.99%) 3786.33

0.00 1.82 (0.07%) 2460.67

Hands-on: NPB-MZ-MPI / BT

