The POP Centre of Excellence

Jon Gibson, NAG

jon.gibson@nag.co.uk

EU H2020 Centre of Excellence (CoE)

1 December 2018 – 30 November 2021

Grant Agreement No 824080

The POP CoE

- One of ten EU Centres of Excellence in HPC Applications
  - POP is a CoE in **Performance Optimisation and Productivity**
    - Promoting **best practices in parallel programming**
  - The other CoEs are BioExcel-2, ChEESE, CompBioMed, E-CAM, EoCoE-II, ESiWACE2, EXCELLERAT, HiDALGO and MaX2.

- POP provides **FREE** services
  - for (EU) academic and industrial (parallel) codes and users
  - across all application areas, platforms, scales

- giving users
  - a precise understanding of application and system behaviour
  - suggestions/support on how to refactor code in the most productive way

Motivation

- Time is money – especially on supercomputers
- To run bigger and/or more complex simulations
- To remain competitive with similar codes
- We need to understand the behaviour of a code in order to guide the optimisation process
- Understanding performance is hard
  - Scientific Codes often developed by many people with development driven by functionality rather than performance.
  - HPC machines have complex architectures
    - Many nodes of multicore processors with an interconnect and filesystem, performing vector operations and having several levels of cache.

The POP Partners

- **Who?**
  - BSC, ES (coordinator)
  - HLRS, DE
  - IT4I, CZ
  - JSC, DE
  - NAG, UK
  - RWTH Aachen, IT Center, DE
  - TERATEC, FR
  - UVSQ, FR

A team with

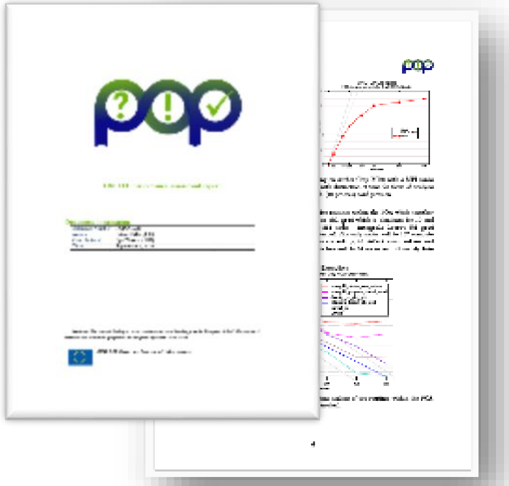
- Excellence in performance tools and tuning
- Excellence in programming models and practices
- A research and development background and a proven commitment to real academic and industrial applications

POP Tools

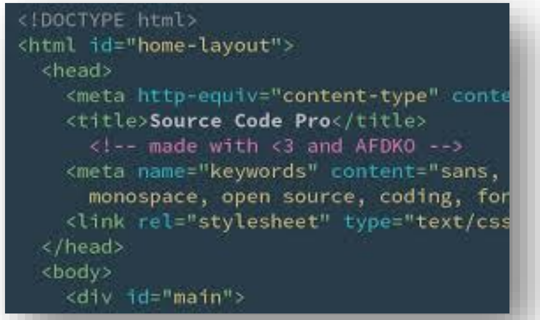
- A number of profiling tools are developed by POP partners
  - BSC Tools – Extrae, Paraver and Dimemas
  - Score-P and Scalasca
- Further development of these tools will take place as part of POP, with a view to improving usability.
- The POP website is a good source of documentation on these and other useful performance tools
  - https://pop-coe.eu/further-information/learning-material

FREE Services provided by the CoE

- **Parallel Application Performance Assessment**
  - Primary service
  - Identifies performance issues of customer code
  - If needed, identifies the root causes of the issues found and qualifies and quantifies approaches to address them (recommendations)
  - 1-3 months effort



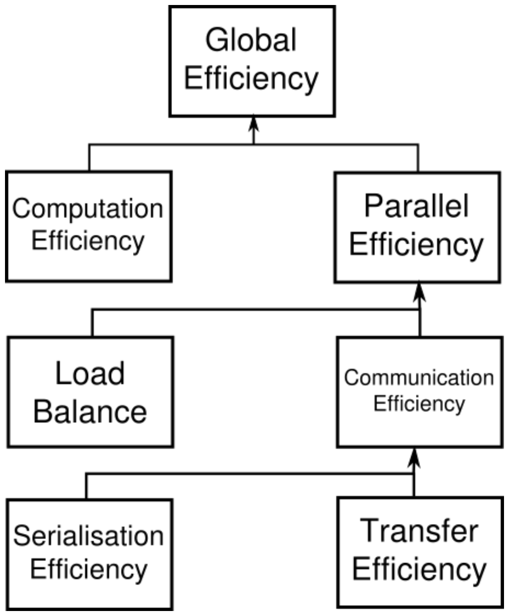
- **Proof-of-Concept**
  - Follow-up service
  - Experiments and mock-up tests for customer codes
  - Kernel extraction, parallelisation, mini-app experiments to show effect of proposed optimisations
  - 3-6 months effort



Note: Effort shared between our experts and customer!

POP Metrics

- The following metrics are used in a POP Performance Audit:
- Global Efficiency (GE): GE = PE * CompE
  - Parallel Efficiency (PE): PE = LB * CommE
    - **Load Balance** Efficiency (LB): LB = avg(CT)/max(CT)
    - **Communication** Efficiency (CommE): CommE = SerE * TE
      - Serialization Efficiency (SerE):
        SerE = max (CT / TT on ideal network)
      - Transfer Efficiency (TE): TE = TT on ideal network / TT
  - (Serial) **Computation** Efficiency (CompE)
    - Computed out of IPC Scaling, Instruction Scaling and Frequency Scaling
    - For strong scaling: ideal scaling -> efficiency of 1.0
- Details see https://sharepoint.ecampus.rwth-aachen.de/units/rz/HPC/public/Shared%20Documents/Metrics.pdf



CT = Computational time
TT = Total time

POP Metrics Example

	MPI Processes			
	6	12	24	48
Global Efficiency	**92%**	**95%**	**62%**	**49%**
Parallel Efficiency	92%	89%	50%	40%
Load Balance	95%	94%	88%	88%
Communication Efficiency	96%	95%	57%	45%
Serialisation Efficiency	97%	96%	76%	62%
Transfer Efficiency	99%	99%	75%	73%
Computational Scalability	100%	106%	124%	123%
IPC Scalability	100%	106%	127%	130%
Instruction Scalability	100%	99%	98%	96%
Frequency Scalability	100%	101%	97%	95%

Success Stories

Some Success Stories

- See ⇨ https://pop-coe.eu/blog/tags/success-stories

Performance Improvements for SCM's ADF Modeling Suite

3x Speed Improvement for zCFD Computational Fluid Dynamics Solver

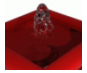
25% Faster time-to-solution for Urban Microclimate Simulations

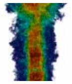
2x performance improvement for SCM ADF code

Proof of Concept for BPMF leads to around **40% runtime reduction**

POP audit helps developers **double their code performance**

10-fold scalability improvement from POP services


POP performance study improves performance **up to a factor 6**

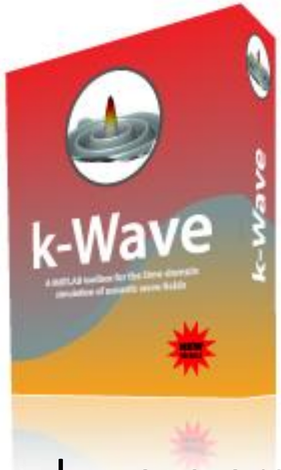

POP Proof-of-Concept study leads to **nearly 50% higher performance**


POP Proof-of-Concept study leads to **10X performance improvement** for customer

Improvements

Reductions

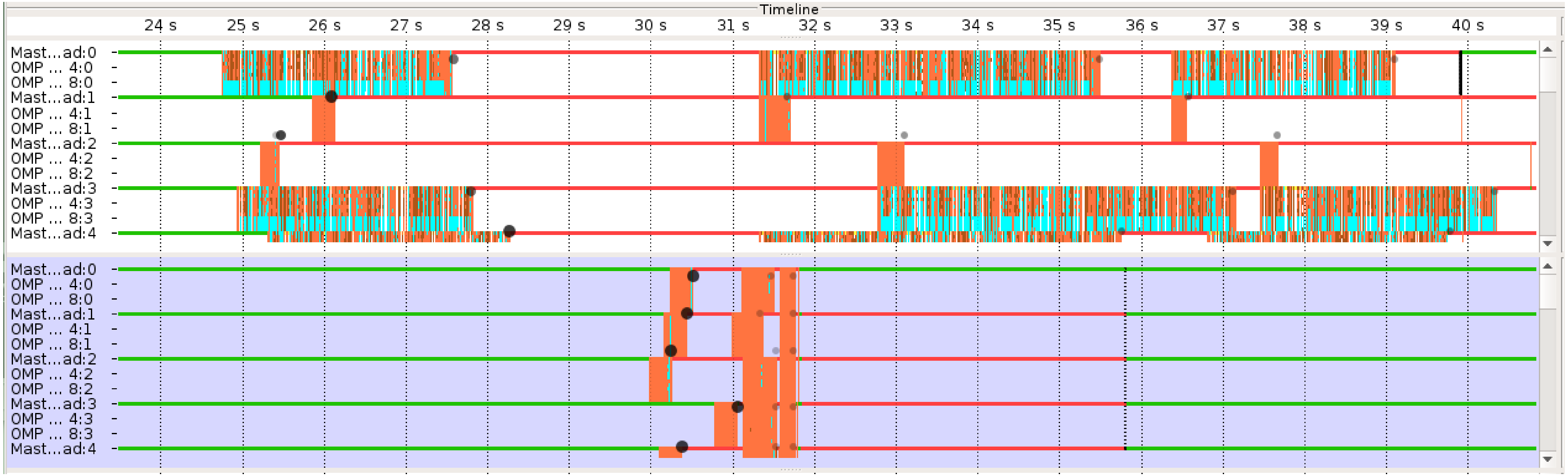
k-Wave – Brno Uni. of Technology



www.k-wave.org

- Toolbox for time domain acoustic and ultrasound simulations in complex and tissue-realistic media
- C++ code parallelised with Hybrid MPI and OpenMP (+ CUDA)
- Profiling showed that
  - 3D domain decomposition suffered from major load imbalance: exterior MPI processes, with fewer grid cells, took much longer than interior
  - OpenMP-parallelised FFTs were much less efficient for grid sizes of exterior, requiring many more small and poorly-balanced parallel loops
- Using a periodic domain with identical halo zones for each MPI rank reduced the overall runtime by a factor of 2.

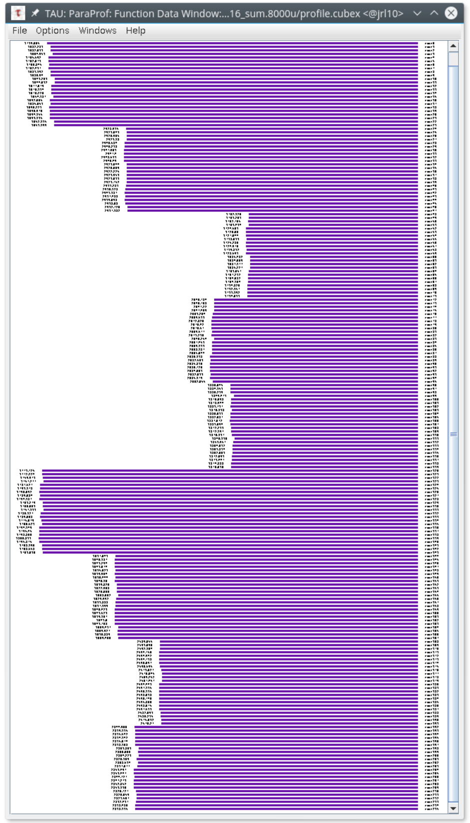
k-Wave – Brno Uni. of Technology



- Comparison time-line before (top) and after (bottom) balancing, showing exterior MPI ranks (0,3) and interior MPI ranks (1,2)
  - MPI synchronization in red; OpenMP synchronization in cyan

EPW – University of Oxford

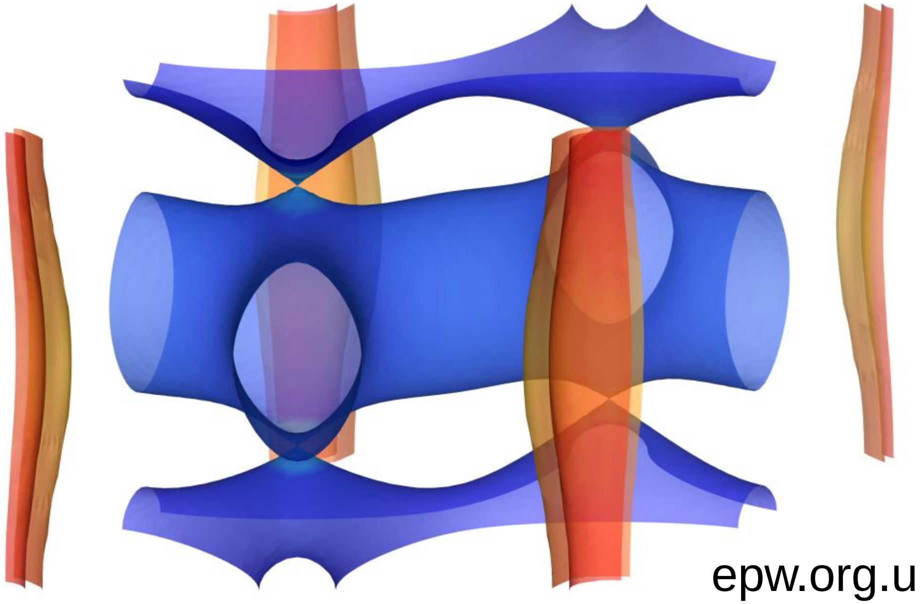
- Electron-Phonon Wannier (EPW) materials science DFT code; part of the Quantum ESPRESSO suite
- Fortran code parallelised with MPI
- Profiling showed
  - Poor load balance
  - Large variations in runtime, likely caused by I/O
  - Final stage spends a great deal of time writing output to disk



EPW – University of Oxford

- Original code had all MPI processes writing result to disk at the end.
- This was modified this so that only one process wrote the output.
- On 480 MPI processes, time taken to write results fell from over 7 hours to just 56 seconds: a 450-fold speed-up!

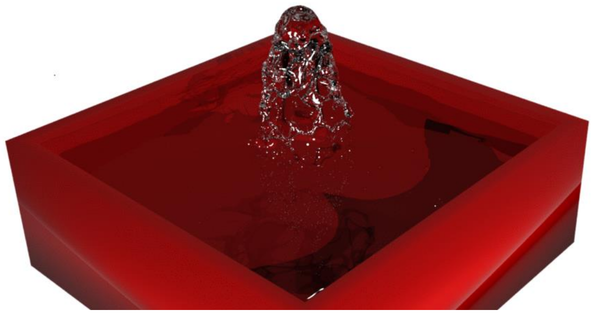
- Combined with other improvements, this enabled simulations to scale to a previously impractical 1920 MPI processes.



epw.org.uk

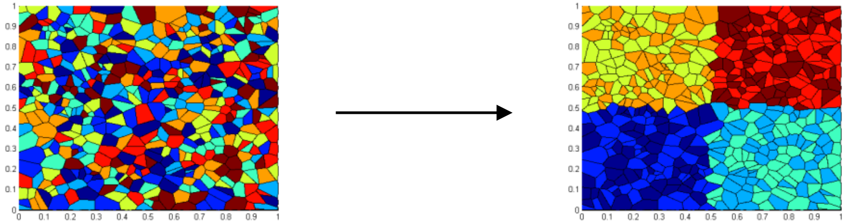
sphFluids – Stuttgart Media University

- Smoothed particle hydrodynamics code
  - C++ with OpenMP
- Profiling identified several issues
  - Definitions of variables in inner loops
  - Unnecessary operations caused by indirection in code design
  - Frequently-used non-inlined functions
  - High cache misses, which could be reduced by reordering the processing of particles
- The developers decided to completely rewrite the code based on their new knowledge, leading to an overall performance improvement of 5x - 6x.



GraGLeS2D – RWTH Aachen

- Simulation of microstructure evolution in polycrystalline materials
- After profiling, the following optimisations were implemented
  - Memory allocation library optimised for multi-threading
  - Reordering the work distribution to threads



  - Algorithmic optimisation in the convolution calculation
  - Code restructuring to enable vectorisation
- An improvement of over 10x was demonstrated for the region concerned, with an overall application speed-up of 2.5x.

Dissemination and Contact

Accessing POP Services

- If you're a code developer or user interested in a free performance assessment of a code, you can sign up to the service directly via the POP website.
  - Feel free to contact us first to discuss the service and what might be possible.
- Alternatively, if you're part of a service with a number of candidate codes on your systems, we'd be happy to discuss how we might work together.
- If you're hosting or know of any events which we could attend to inform people about our services, then let us know.

POP Webinars

- The service runs regular webinars on topics of interest to the community
  - Profiling and performance assessment and optimisation
  - Aspects of parallel programming
  - Let us know what you'd like us to cover!
- 30 minute presentations, then attendees' questions answered
- Recordings and the slides from all previous webinars can be found at https://pop-coe.eu/blog/tags/webinar

Keep in touch!

- Browse the POP website at https://pop-coe.eu
  and subscribe to the newsletter.
- Follow us on twitter @POP_HPC
- Subscribe to the POPHPC YouTube Channel
- Join the LinkedIn group

Performance Optimisation and Productivity

A Centre of Excellence in HPC

Contact:

https://www.pop-coe.eu

mailto:pop@bsc.es

@POP_HPC

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 676553 and 824080.