An Introduction to the POP Centre of Excellence

Jon Gibson

EU H2020 Centre of Excellence (CoE)

1 December 2018 – 30 November 2021

Grant Agreement No 824080

Contents

- The POP Centre of Excellence
- The Need for Performance Profiling
- The POP Tools
- The POP Metrics
- How to make use of POP Services

The POP Centre of Excellence

- POP is a CoE in **Performance Optimisation and Productivity**
  - Promoting **best practices in parallel programming**
- POP provides **FREE** services
  - for (EU) industrial and academic, (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
- often leading to
  - faster code / bigger jobs / better science
  - a significant Return on Investment
  - an edge against the competition

Return on Investment Examples

Application Savings after POP Proof-of-Concept

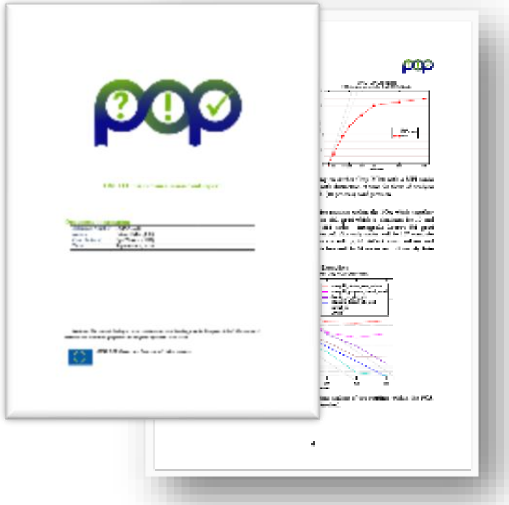
- POP PoC resulted in 72% faster-time-to-solution
- Production runs on ARCHER (UK national academic supercomputer)
- Improved code saves €15.58 per run
- Yearly savings of around €56,000 (from monthly usage data)

Application Savings after POP Performance Assessment

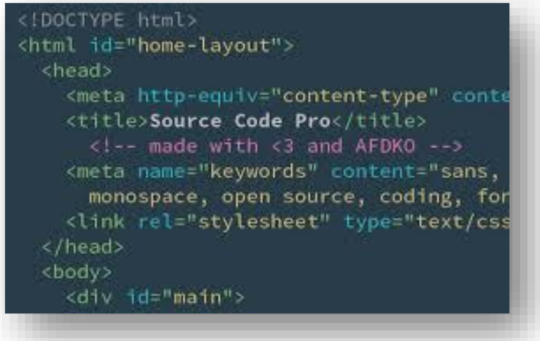
- Cost for customer implementing POP recommendations: €2,000
- Achieved improvement of 62%
- €20,000 yearly operating cost
- Resulted in yearly saving of €12,400 in compute costs ⇨ ROI of 620%

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



```
<!DOCTYPE html>
<html id="home-layout">
  <head>
    <meta http-equiv="content-type" conte
    <title>Source Code Pro</title>
      <!-- made with <3 and AFDKO -->
    <meta name="keywords" content="sans,
      monospace, open source, coding, for
    <link rel="stylesheet" type="text/css
  </head>
  <body>
    <div id="main">
```

Note: Effort shared between our experts and customer!

The Need for Performance Profiling

- 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.
- We need to be very selective before spending time optimising code
  - *"Premature optimization is the root of all evil."* – Donald Knuth
  - Optimising code is often time-consuming
  - Optimised code is often more difficult to read/understand (hence debug/maintain)
  - Optimising a routine that only takes 2% of the execution time is going to have very little impact on the overall performance
- We therefore need a way to understand the behaviour of a code in order to guide the optimisation process

Performance Profiling

- *Profiling* refers to the monitoring of a code's behaviour as it executes
  - To capture the behaviour of the application under production conditions
  - To understand and quantify the efficiency of resource usage
  - To identify inefficiencies and where improvements can be made
- These profiling results
  - Guide the code refactoring effort
  - Provide a baseline from which improvements can be measured

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

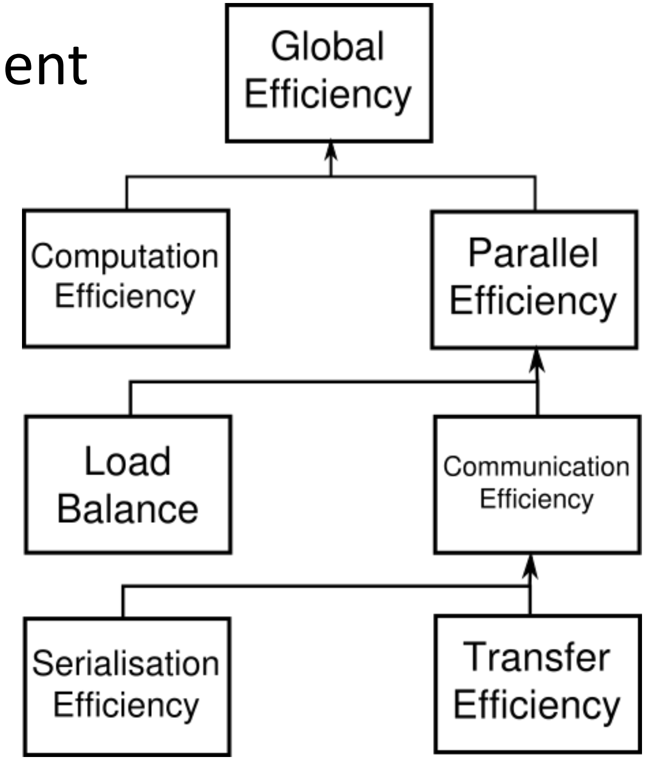
- Excellence in performance tools and tuning
- Excellence in programming models and practices
- A research and development background and a proven commitment to academic and industrial applications

POP Tools

- A number of profiling tools are developed by POP partners
    - Extrae, Paraver and Dimemas
    - Score-P, Cube and Scalasca
    - MAQAO
    - PyPOP
- Further development of these tools is currently taking place as part of POP, with a view to improving usability.
- Some of the talks in this series cover the installation and use of these tools.

POP Metrics

- The following metrics are used in a POP Performance Assessment
- Global Efficiency
  - Parallel Efficiency
    - **Load Balance** Efficiency
    - **Communication** Efficiency
      - Serialization Efficiency
      - Transfer Efficiency
  - (Serial) **Computation** Efficiency
    - Computed out of IPC Scaling, Instruction Scaling and Frequency Scaling
    - For strong scaling: ideal scaling -> efficiency of 1.0
- More details in the next talk.

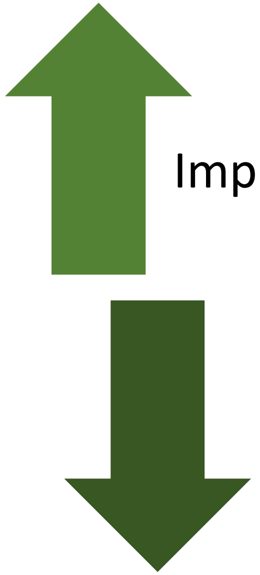


POP Metrics Example

	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%

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

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: https://pop-coe.eu/request-service-form
  - 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.

POP Online Training

1. An Introduction to the POP Centre of Excellence
2. Understanding Application Performance with the POP Metrics
3. Computing the POP Metrics with PyPOP
4. Installing POP Tools – Extrae and Paraver
5. Installing POP Tools – Scalasca and Cube
6. Introduction to Extrae and Paraver
7. Introduction to Scalasca and Cube

POP Webinars

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

Performance Optimisation and Productivity

A Centre of Excellence in HPC

Contact:

- https://www.pop-coe.eu
- pop@bsc.es
- @POP_HPC
- youtube.com/POPHPC

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