



Determining Parallel Application Execution Efficiency and Scaling using the POP Methodology

Marta Garcia-Gasulla, Barcelona Supercomputing Center, marta.garcia@bsc.es

Sandra Mendez, Barcelona Supercomputing Center, sandra.mendez@bsc.es

Anke Visser, Jülich Supercomputing Centre, a.visser@fz-juelich.de

Brian Wylie, Jülich Supercomputing Centre, b.wylie@fz-juelich.de

This guide provides an overview of the tutorial at the ISC-HPC'24 conference¹, so that attendees know what to expect and how to prepare accordingly.

1 Tutorial Overview

This tutorial presents the methodology developed and applied over several years within the HPC Centre of Excellence Performance Optimisation and Productivity (POP)². Its focus is the hierarchy of execution efficiency and scaling metrics that identify the most critical issues and quantify potential benefits of remedies. The metrics can be readily compared and determined by a variety of tools for applications in any language employing standard MPI, OpenMP and other multi-threading, tasking and offload paradigms. Widely-deployed open-source tools will be used to demonstrate this process with provided performance measurements of actual HPC application executions, allowing tutorial participants to repeat this on their own computers and preparing them to locate and diagnose efficiency and scalability issues in their own parallel application codes.

1.1 Format

The half-day tutorial consists of sessions of 120 and 90 minutes, separated by a 30 minute break. Questions from participants are encouraged throughout the tutorial, as well as beforehand and afterwards via E-mail.

To complement the presentations, the performance analysis process will be demonstrated with various tools and several sample measurements. The sample execution profiles and traces are available to download for those who wish to follow along with tools installed on their own desktop/notebook computers.

1.2 Requirements

A general understanding is expected of how HPC applications are programmed and run using MPI message-passing and/or OpenMP multithreading, tasking and offload of kernels.

It is **not** expected that attendees are already familiar with the featured performance analysis tools, particularly their associated instrumentation and measurement collection tools, however, this is certainly advantageous and highly recommended. Recordings and presentations from recent VI-HPS workshops cover the use of performance tools from BSC and JSC.

¹<https://app.swapcard.com/event/isc-high-performance-2024/planning/UGxhbm5pbmdfMTgyNTYzMw==>

²<https://www.pop-coe.eu/>



After the tutorial participants are expected to be able to undertake the presented analyses with their own execution profiles and traces obtained from execution measurements of their own parallel applications. For tutorial participants who already have such measurements, it may be possible for instructors to examine them with you if time permits.

1.3 Preparation

Tools installed on remote computer systems may be used (e.g. on HPC systems at computing centres), however, X11-forwarding may be sluggish for interactive GUI usage. Local installations are therefore strongly recommended. Since downloading and installing the graphical tools (and any dependencies) may take some time, and potentially require assistance, this should be done well in advance of the tutorial session.

1.3.1 BSC tools

The Paraver trace visualizer and analyzer GUI (for Linux, macOS and Windows), along with Extrae instrumentation framework, Dimemas network simulator and associated performance analytics tools, can be downloaded from:

<https://tools.bsc.es/downloads>

Only the Paraver GUI and basic analysis modules will be used for the tutorial, so there's no need to install any of the other tools.

To verify your Paraver GUI installation, load one of the provided '.prv' files from the lulesh subdirectory, e.g. `lulesh/lulesh2.0_i_27p.prv`

For assistance, contact tools@bsc.es

1.3.2 JSC tools

A POP online training module covers installation of Scalasca/Score-P and CUBE on Linux computer systems:

<https://pop-coe.eu/further-information/online-training/installing-pop-tools-score-p-scalasca-cube>

While Scalasca/Score-P are used for measurement and analysis, only the analysis report explorer GUI known as CUBE (version 4.8.2 or later) is required to follow along with this tutorial. CUBE binary installation packages for Windows and macOS are available for download:

<https://www.scalasca.org/scalasca/software/cube-4.x/download.html>

To verify your CUBE GUI installation, load one of the provided '.cubex' files from the BT-MZ subdirectory, e.g. `BT-MZ/scorep_bt-mz_B_12x4_sum.def/summary.cubex`

For assistance, contact scalasca@fz-juelich.de