Projects

Performance Dynamics of Massively Parallel Codes

The deployment of adaptive algorithms and sophisticated load-balancing schemes make the execution behavior of simulation programs increasingly dynamic. Effective application optimization therefore requires capturing and analyzing performance data along the dimensions of both space and time. While existing performance analysis tools typically provide detailed information along spatial dimensions like processes and nodes, the aspect of performance dynamics, that is, the evolution of performance metrics along the time dimension, has so far been neglected. Additional insight that can be obtained by taking advantage of performance dynamics is depicted in Fig. 1.

To support the optimization of applications with time-dependent performance characteristics, the LMAC project (Leistungsdynamik massiv-paralleler Codes), funded under the second call of the BMBF program "HPC-Software für skalierbare Parallelrechner", aims to extend the well-established performance-analysis tools Vampir, Scalasca, and Periscope with new functionality to measure and analyze performance dynamics. Vampir is an interactive trace browser whose particular strength is the detailed visualization of the interactions between the different processes of a parallel program, offering highly flexible views to the user. Scalasca, which has been specifically designed for large-scale systems, integrates efficient performance summaries with the ability to automatically identify wait states that occur in simulation codes, for example as a result of unevenly distributed workloads. Whereas the first two tools analyze the performance data postmortem, that is, after the parallel program has been terminated, Periscope characterizes the performance properties of an application and quantifies associated overheads already at runtime.

A major portion of the new capabilities will be integrated in Score-P, a measurement infrastructure shared by all of the above-mentioned tools, which was created in the SILC project (2009-2011) funded under the first call of the same BMBF program. Score-P was designed for scalability and easy of use, supports profiling, event tracing, and online analysis of HPC applications. It enables enhanced interoperability between the end-user tools and will soon replace their proprietary measurement systems (see Fig. 2). The organization of Score-P as a community project is another task in LMAC.

To serve a broad user base, both, within the Gauss Alliance and beyond, most of the performance tools are released free of charge to the community under an open-source license. Only Vampir, due to its sophisticated user interface, is distributed commercially. The software products are accompanied by training- and support offerings through the Virtual Institute – High Productivity Supercomputing, and will be maintained and adapted to emerging HPC architectures and programming paradigms beyond the original project duration.

The academic project partners in LMAC are the German Research School for Simulation Sciences as the coordinator, the Jülich Supercomputing Centre, RWTH Aachen University, TU Dresden and TU Munich. GNS mbH, a private company that specializes in services related to metal forming simulations, such as mesh generation for complex structures and finite element analyses, joined the project as an industrial partner. In addition, the University of Oregon, an associated partner, complements the LMAC objectives with corresponding extensions to the performance tool TAU.

For more information see:
http://www.vi-hps.org/projects/lmac
and http://www.score-p.org

- Christian Rössel [1]
- Bernd Mohr [1]
- Michael Gerndt [2]
- Felix Wolf [3]

[1] Jülich Supercomputing Centre (JSC)

[2] Technische Universität München

[3] German Research School for Simulation Sciences, RWTH Aachen

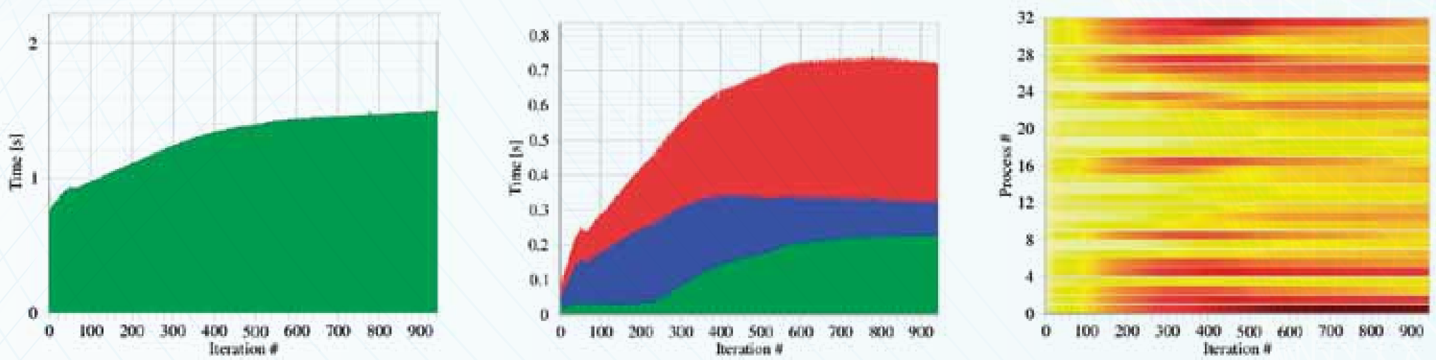

Figure 1: Runtime-distribution and point-to-point communication times over iterations and processes: (a) Runtime, (b,c) Point-to-point communication. The different colors in (b) correspond to the minimum, the median, and the maximum over all processes of a single iteration. In (c) the color coding denotes the entire distribution over space-time (darker means more). The rise in the maximum communication time in (b) is the cause for the increased iteration-runtime in (a).

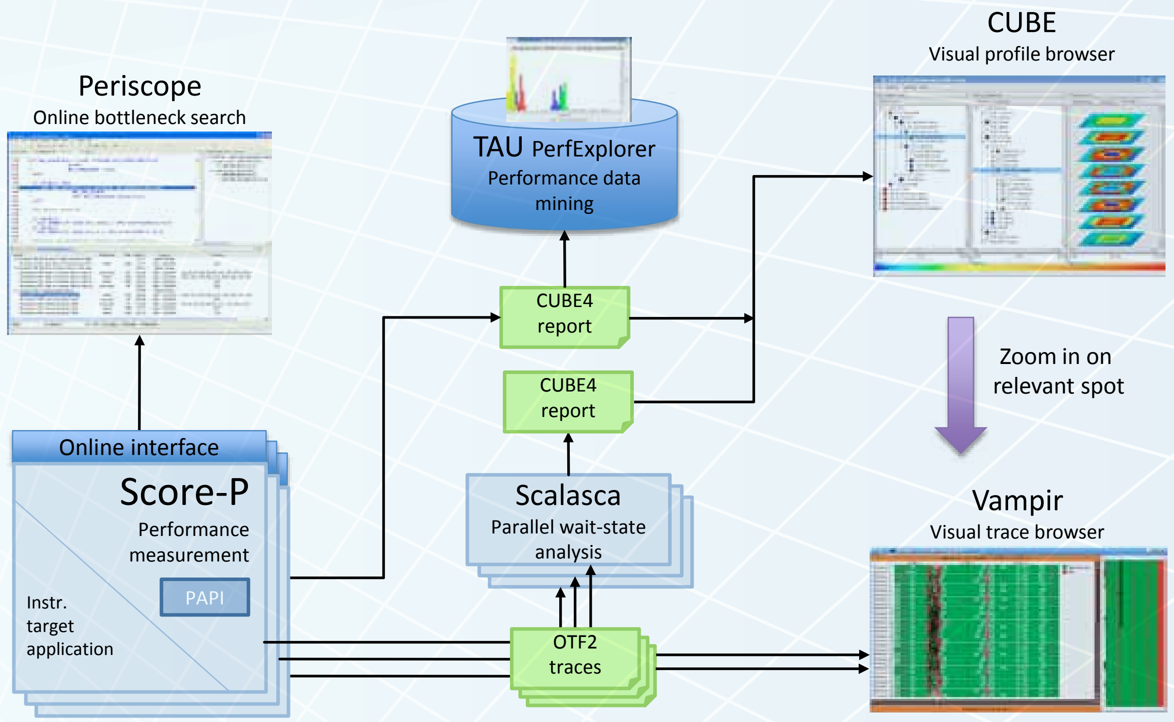

Figure 2: Interaction of the tools Persicope, Scalasca, Vampir, TAU and Scalasca's profile browser CUBE via the data exchange formats OTF2 for traces and CUBE4 for profiles.