Improved Support of One-sided Communication Libraries in the Scalasca Toolset

MARC-ANDRÉ HERMANNS

The Scalasca toolset (1), developed since 2006 by the Jülich Supercomputing Centre and since 2009 jointly developed with the German Research School for Simulation Sciences, supports scalable performance analysis of large-scale parallel applications using MPI and OpenMP on a wide range of HPC systems. It offers runtime summarization as well as event tracing through direct measurement on instrumented executables. Research activities involve new measurement techniques as well as the support of emerging programming paradigms.

Partitioned global address space (PGAS) languages (2) provide a global shared-memory abstraction for shared- and distributed-memory computer architectures. They are designed to enable application developers to express distributed algorithms more clearly and directly than with explicit communication libraries, increasing maintainability and overall productivity on the emerging extreme-scale high-performance computing (HPC) systems, comprising hundreds of thousands of processing elements. In contrast to the fine-grained control of explicit communication with the Message Passing Interface (3) (MPI), the defacto standard for programming distributed-memory computer systems, PGAS languages provide the user with virtual direct access to remote data. They handle these remote accesses using one-sided communication libraries, such as SHMEM, ARMCI, GASNET, or the newly designed DMAPP on the Cray XE systems.

One of the tasks for the HPC-Europa2 visit to EPCC was the implementation of a measurement adapter for the SHMEM one-sided communication library, available of EPCC's Cray XT system HECToR. SHMEM supports a profiling interface similar to the PMPI interface known from MPI that allows interposing additional measurement code by simply linking it to the application when the executable is built. Unfortunately, SHMEM—unlike MPI—is not standardized and exists in slightly different variations on several platforms. To handle this well, the internal Scalasca tool to generate measurement adapters was modified to support different variants of the SHMEM interface.

Some experts suggest that a hybrid approach of MPI and another thread-based programming model in a single application is well suited to program the extreme-scale HPC systems of the future, comprising tens or even hundreds of thousands of processing elements. Others advocate that using two orthogonal programming paradigms like MPI and OpenMP together may raise the programming complexity to a level not manageable by the average simulation developer. As a result parallel efficiency of the simulations may not turn out to be optimal and overall productivity may drop.

Currently, many application developers who use PGAS languages for their simulation code struggle to achieve the same performance as with a pure MPI version. We see the reasons for this in (1) a lack of experience of the application developer in how to tune their PGAS codes, (2) insufficient support through software development tools, and (3) crucial information relevant to performance being lost due to the abstraction, resulting in sub-optimal output by the compiler.

Low-level performance data can be obtained by the before-mentioned work on analyzing the communication layer. However, PGAS languages present themselves to the user with a shared-memory view of the data, thus performance problems on this higher level cannot be expressed in terms of communication, as there is no explicit communication from the users point of view. The terminology here will have to be closer to that of shared-memory computers (e.g., cache misses), where performance is largely dominated by synchronization overheads, data locality, caching behavior and bulk data operations (i.e., vectorization). The second half of the visit to EPCC was spent on discussions with the EPCC experts on shared-memory programming about how to express performance problems on a high abstraction level and how to obtain the relevant data.

As a direct result of the HPC-Europa2 visit to EPCC, experimental measurement support of Cray SHMEM as well as GASNET driven applications with the Scalasca toolset was achieved. Furthermore, discussions with the local experts laid the foundation for future developments in formulating performance metrics of PGAS languages.

Acknowledgements. The work has been performed under the HPC-EUROPA2 project (project number: 228398) with the support of the European Commission – Capacities Area – Research Infrastructures.

References

(1) http://www.scalasca.org/
(2) http://www.pgas.org/
(3) http://www.mpi-forum.org/

German Research School for Simulation Sciences GmbH, Germany