Exascale MPI / MPICH

Efficient communication among the compute elements within high performance computing systems is essential for simulation performance. The Message Passing Interface (MPI) is a community standard developed by the MPI Forum for programming these systems and handling the communication needed. MPI is the de facto programming model for large-scale scientific computing and is available on all the large systems; most of DOE's parallel scientific applications running on pre-exascale systems use MPI. The goal of the Exascale-MPI project is to both evolve the MPI standard to fully support the complexity of the exascale systems and deliver MPICH, a reliable, performant implementation of the MPI standard, for these systems.

While MPI will continue to be a viable programming model on exascale systems, both the MPI standard and the MPI implementations need to address the challenges posed by the increased scale, performance characteristics, evolving architectural features, and complexity expected from the exascale systems as well as provide support for the capabilities and requirements of the applications that will run on these systems.

Therefore, this project addresses five key challenges to deliver a performant MPICH implementation: (1) scalability and performance on complex architectures that include, for example, high core counts, processor heterogeneity, and heterogeneous memory; (2) interoperability with intranode programming models having a high thread count such as OpenMP, OpenACC, and emerging asynchronous task models; (3) software overheads that are exacerbated by lightweight cores and low-latency networks; (4) extensions to the MPI standard based on experience with applications and high-level libraries and frameworks targeted at exascale; and (5) topics that become more significant for exascale architectures—memory and power usage, and resilience.

The MPICH development effort continues to address several key challenges such as performance and scalability, heterogeneity, hybrid programming, topology awareness, and fault tolerance. Several additional features are being developed in order to support the exascale machines that will be deployed, including (1) support for multiple accelerator modes and native hardware models that will facilitate data transfers between GPU accelerators and the communication network in cases where native hardware support is lacking and (2) offline and online performance tuning based on static and dynamic system configurations, respectively.

This team will also produce a significantly larger test suite to stress test various use cases of MPI and develop a test generation toolkit that automatically profiles MPI usage by applications (using the MPI profiling interface) and generates a simple test program that represents the MPI communication pattern of the application, covering basic MPI features, sanitized iterative loops, memory buffer management, and incomplete executions. These activities will help improve both the reliability and performance of the MPICH implementation and other MPI implementations as they evolve.

The team will continue to engage with the MPI Forum to ensure that future MPI standards meet the needs of both the ECP and broader DOE applications. To achieve good performance on exascale machines, the team plans to develop new MPI features for application-specific requirements, such as alternative fault tolerance models and reduction neighborhood collectives, either through the inclusion in the standard or as extensions to the standard.

Progress to date

- The Exascale-MPI team developed a high-performance, production-quality MPI implementation called MPICH. The team continues to improve the performance and capabilities of the MPICH software in order to meet the demands of ECP and other broader DOE applications.

- Some technical risks that have been retired include scalability and performance over complex architectures and interoperability with intranode programming models having high thread count such as OpenMP.

PI: Pavan Balaji, Argonne National Laboratory

Collaborators: Argonne National Laboratory