

DOE-SUPPORTED COMPUTING TECHNOLOGIES THAT MADE A DIFFERENCE

MPI AND MPICH

RUSTY LUSK

Mathematics and Computer Science Division
Argonne National Laboratory

September 26, 2017

MPI IS 25 YEARS OLD THIS YEAR!

- Anniversary symposium at Argonne yesterday in conjunction with EuroMPI/USA 2017 conference



- Web page: <http://www.mcs.anl.gov/mqi-symposium>

OUTLINE

- What is MPI?
 - Brief history of the Message Passing Interface standard
- What is MPICH?
 - Brief history of an influential implementation of MPI
- Role of ASCR support
 - Prehistory of MPI and MPICH: ASCR support for parallel computing research (particularly at Argonne)
 - Golden age of ASCR support for MPI/MPICH
 - The tight spot
 - Today
 - The MPI Forum
 - MPI/MPICH in the Exascale Computing Project
- Conclusions

WHAT IS MPI?

- MPI is a message passing interface library specification, designed to become an industry standard
 - Deals primarily with movement of data between and among address spaces, either 2-sided (send/receive) or 1-sided (put/get), and collective operations
 - A specification, not a particular implementation
 - A library, not a language (you link to it, rather than compile it)
 - Not an official standard (like IEEE or ISO), so acceptance is entirely based on how useful it is
- Goals: Portability, performance, expressiveness
 - Not especially ease of use (left to libraries)

BRIEF HISTORY OF MPI

- In the late 80s, vendors of commercial parallel computers used their programming models and systems as part of their competition with one another, so parallel programs written for one system could not be run on another.
- A number of research systems were layered over these to provide portability.
- The MPI Forum assembled in 1992 to create the definition of a portable library interface.
 - There had also been some earlier individual efforts in this direction.
- The Forum was composed of computer scientists, vendor representatives who recognized the need to expand the market, and application computational scientists.
- It was an open process; anyone could attend meetings (every 6 weeks in a lousy hotel in North Dallas); minutes and drafts were publicly available.

BRIEF HISTORY OF MPI (CONT.)

- There have been multiple phases of the standardization effort:
 - MPI-1 ('93-'94): basic send/receive and collectives, communicators and datatypes, Fortran (-77) and C bindings
 - MPI-2 ('95-'97): 1-sided, parallel I/O, dynamic process management, F90 & C++ bindings, basic thread safety
 - MPI-3 ('10-'12): new 1-sided, non-blocking and neighborhood collectives, shared memory, more on interactions with threads, C++ out
 - MPI-4 ('13 -...): threads as “end points,” fault tolerance, sessions, persistence for collectives, ...?
- A current MPI implementation is now taken for granted by applications on all parallel computing systems.

BRIEF HISTORY OF MPICH

- MPICH is one implementation of the MPI standard that has “made a difference.”
- When the Forum was first formed, the Argonne participants (Bill Gropp and Rusty Lusk) undertook to provide an immediate implementation.
- This was possible because each of us had already developed general-purpose portable parallel libraries (p4 and Chameleon).
- We called it MPICH (the CH is for Chameleon).
- MPICH tracked the evolving specification, which changed every six weeks.
- This effort supported the standard definition effort in two ways:
 - It provided immediate feedback to the Forum on implementation issues that were sometimes overlooked at the initial design level.
 - When the Forum released the Standard after 18 months, a complete, open-source, portable implementation was available to users and vendors.
- MPICH was architected in such a way that vendors could modify relatively small sections of the code to adapt it for high performance on their individual products.

HISTORY OF MPICH (CONT.)

- MPICH also played (and continues to play) an important role in our parallel computing research.
- More than 150 peer-reviewed papers from ANL group alone
 - Many more from parallel computing research groups around the world who used MPICH in their research
- Around 2000 we undertook a complete rewrite of MPICH
 - To implement the MPI-2 standard
 - To incorporate lessons learned from experience with first MPICH
 - To modularize it with internal interfaces to support research by ourselves and others, as well as alternate implementations of subsystems by vendors
- R&D 100 Award, 2005
- MPICH is still current (MPI-3) and the basis of both research and commercial implementations
 - About 500,000 lines of C
 - Many 100's of users, providing a wide variety of use cases, feeding into the MPI standardization effort
 - Current on largest, fastest, machines in the world
 - Also used on laptops for development, clusters for testing
 - Large user community self-sustaining (with some help from us)

TODAY: THE MPI FORUM AND COMMUNITY

- The MPI Forum met last week in Chicago
 - 31 attendees from multiple countries, centers, vendors, labs, universities
 - Subcommittees:
 - Tools interface
 - Point to point
 - Persistence
 - Fault Tolerance
 - Sessions
- This week is the annual EuroMPI meeting of the international MPI community
 - (Descended from the original EuroPVM meetings)
 - Happens to be in Chicago this year
 - Yesterday was the “25 Years of MPI” symposium, held as a workshop at the meeting.
 - One conclusion: MPI-related research still needed
 - 16 research papers, on standards issues, implementations, tools.
 - Panel on “post-exascale”

TODAY: MPI AND MPICH IN ECP

- The Argonne group (headed by Pavan Balaji) is funded by an ECP Software Technology grant: “Exascale MPI.”
- Project goals:
 - Enhance MPICH to support new features in future versions of the MPI standard and specific challenges posed by exascale architectures
 - Enhance the MPI standard itself by taking leading roles in the MPI Forum
 - Investigate new programming approaches beyond those in the current MPI standard
 - Interact with vendors involved in DOE supercomputer acquisitions, to codesign MPICH such that it remains the leading MPI implementation running on the fastest machines in the world.
- Specific interactions within ECP
 - Other software projects: OpenMP, UPC++, Tools
 - 6 specific application projects: CEED, Lattice QCD, ACME, HACC, NWCemEx, AMRex
- Collaborations with vendors: Intel, Cray, Mellanox
- ECP also supports the Open MPI implementation of MPI
 - Since MPI is considered a critical component, vendors use both, and for risk reduction.

IMPORTANCE OF SUSTAINED SUPPORT FROM ASCR

- ASCR support for MPI/MPICH (at least at Argonne) was always important, but sometimes more “sustained” than at other times.
- The real impact of sustained DOE support can be traced back well before MPI itself.
- In 1980, ASCR (then MICS) was already supporting work on programming standards: Fortran-88, PL/1, IEEE 754 (Floating point arithmetic).
- In 1983, MICS established what was to become a rich environment for parallel computing research at Argonne:
 - The Advanced Computing Research Facility, which came to house a number of the first commercially available parallel machines
 - Research funding for “Advanced Computer Systems Concepts” (note lack of specificity)
 - This funded multiple researchers figuring out how to program the ACRF machines portably.
 - The ACRF offered free classes in parallel programming to all, to spread the word about parallelism.
- It was out of this environment that p4 and Chameleon emerged, ready to provide MPICH with a running start, which in turn helped establish the MPI standard.

APPROXIMATE FUNDING TIMELINE

- 1983-1992: The ACRF and associated research funding (previous slide)
- 1992- 2010: The “Golden Age” of parallel computing support at Argonne:
 - Reviews rather than competing proposals in response to FOAs
 - Leadership roles in MPI-1 and MPI-2 Forums
 - Significant number of refereed publications on the many issues raised by the standard
 - MPICH, a major software effort, supporting both our research and a user community
 - Community outreach with courses, SC tutorials
 - Collaboration with vendors
 - There was ASCR MPI support at other DOE labs as well.
- 2011-2012: the “tight spot”
 - MPI/MPICH-related funding discontinued
 - MPI did not fit the narrative of “Everything must be completely new for exascale.”
 - “Productivity” became a byword without considering “for whom?”
 - ASCR became averse to what it saw as “software maintenance.”
- 2013: partial funding restored
- 2015: Exascale Project begins, with adequate funding for MPI/MPICH

SUMMARY

- MPI (generously funded indirectly, although not specifically an ASCR project) has enabled portable, high-performance applications at petascale and is expected to be fundamental for exascale applications.
- MPICH (very much an ASCR project) has been crucial to the success of the MPI standardization effort, and is part of vendors' future plans, continuing into exascale.
- The most significant contribution from ASCR was the creation of a flexible research environment from which MPI emerged, followed by continued support for MPI-related research and for MPICH, its most applicable product, as well as the MPI standardization effort in general.