Getting Insider Information via the New MPI Tools Information Interface

EuroMPI 2016

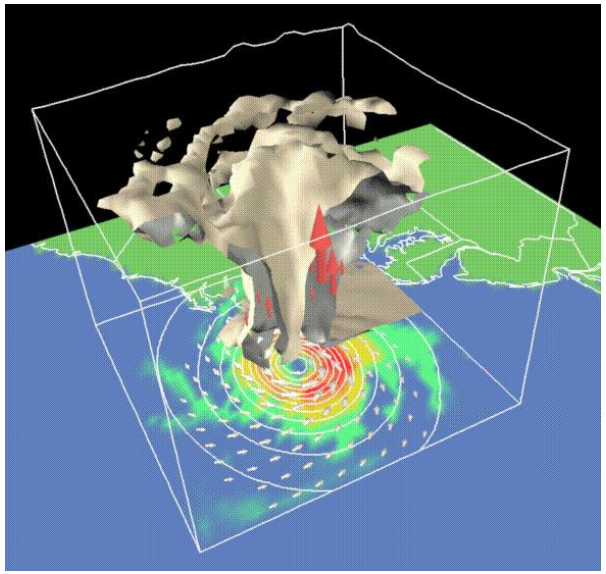
Kathryn Mohror

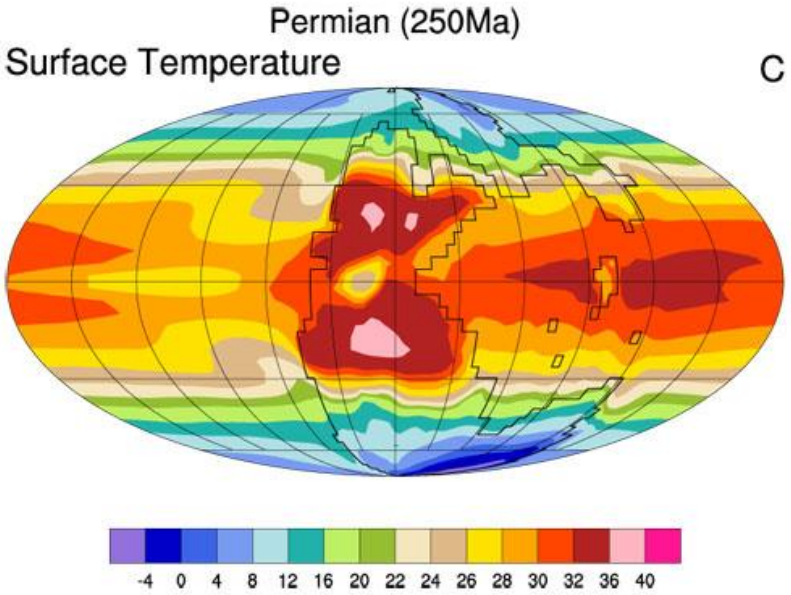
September 26, 2016

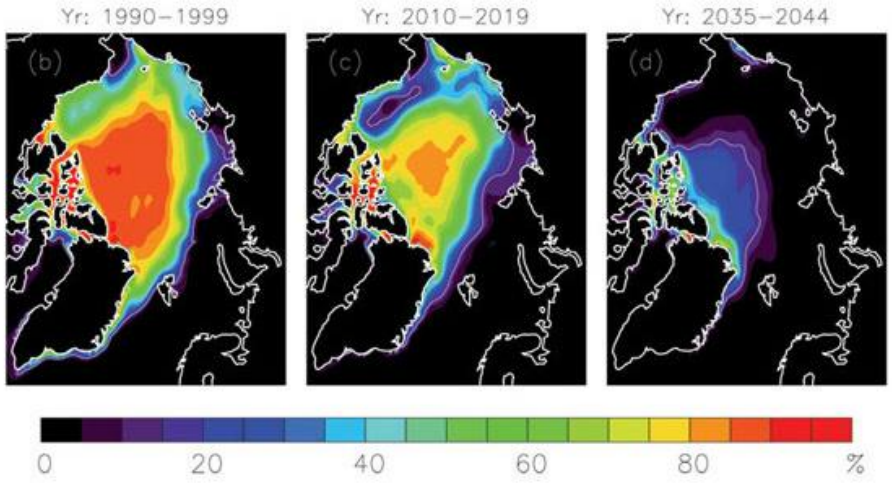
LLNL-PRES-703509
This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under contract DE-AC52-07NA27344. Lawrence Livermore National Security, LLC

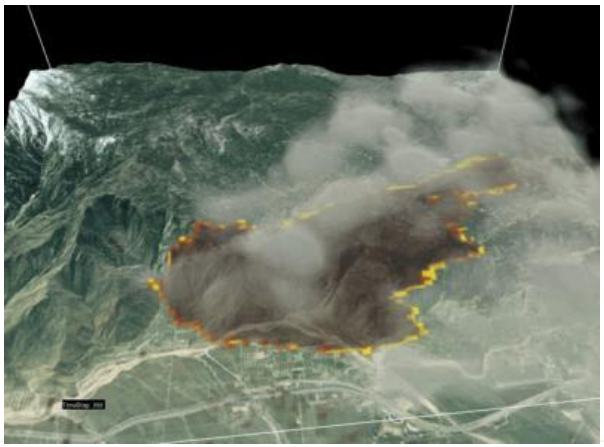
Lawrence Livermore National Laboratory

Applications that run on supercomputers simulate important physical phenomena and we need the answers fast

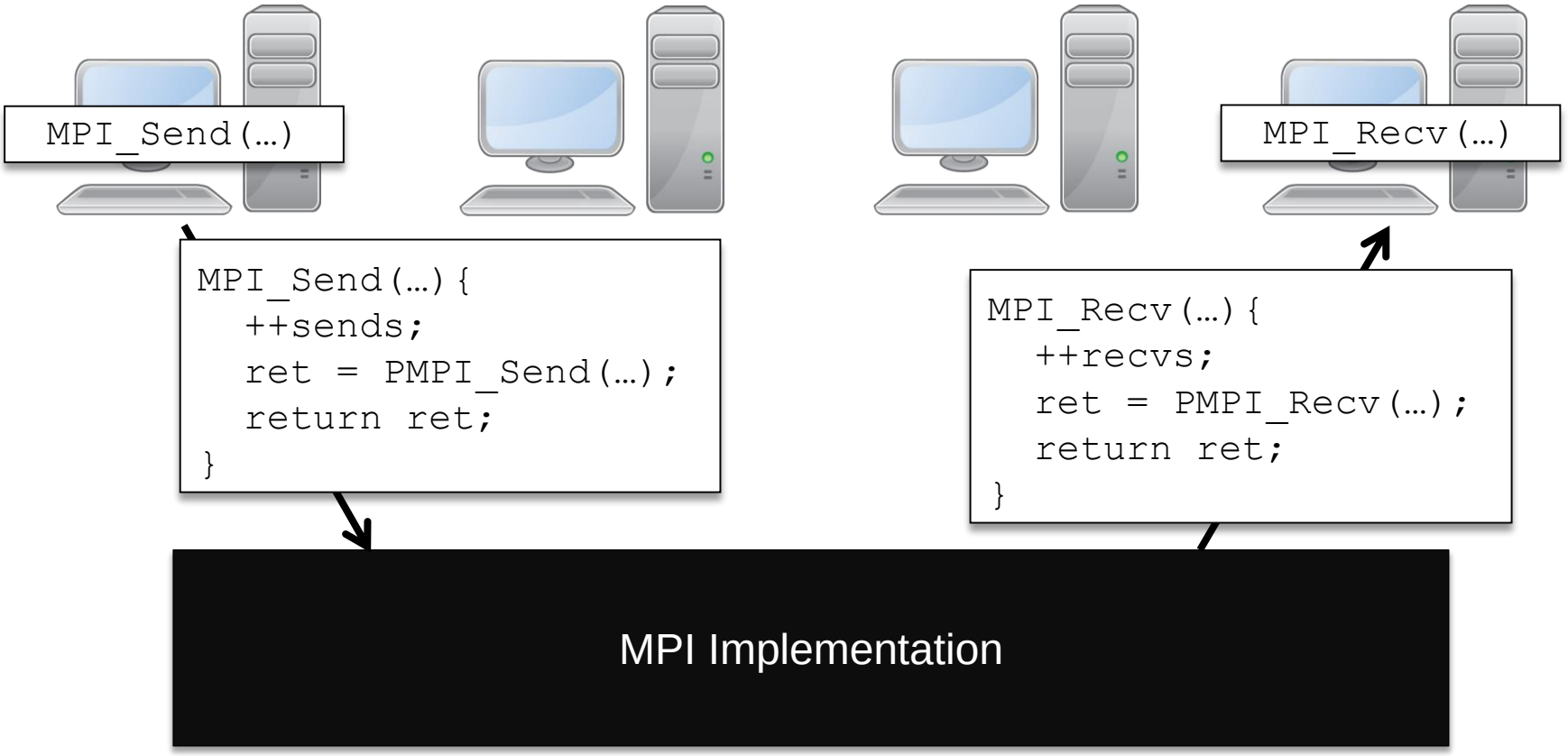






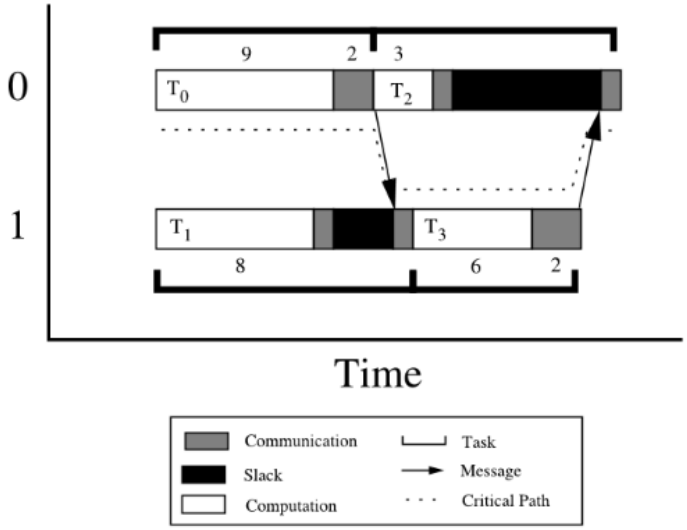


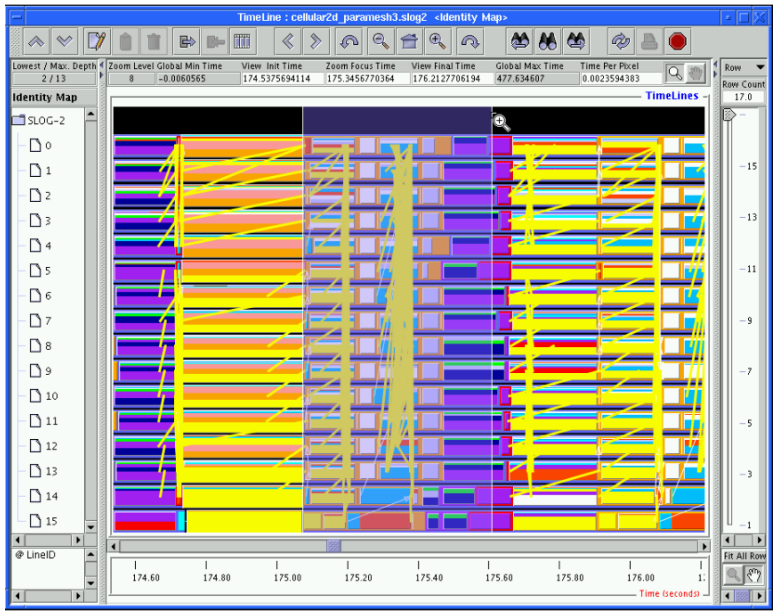
MPI performance analysis tools relied on the profiling interface (PMPI) for 20+ years



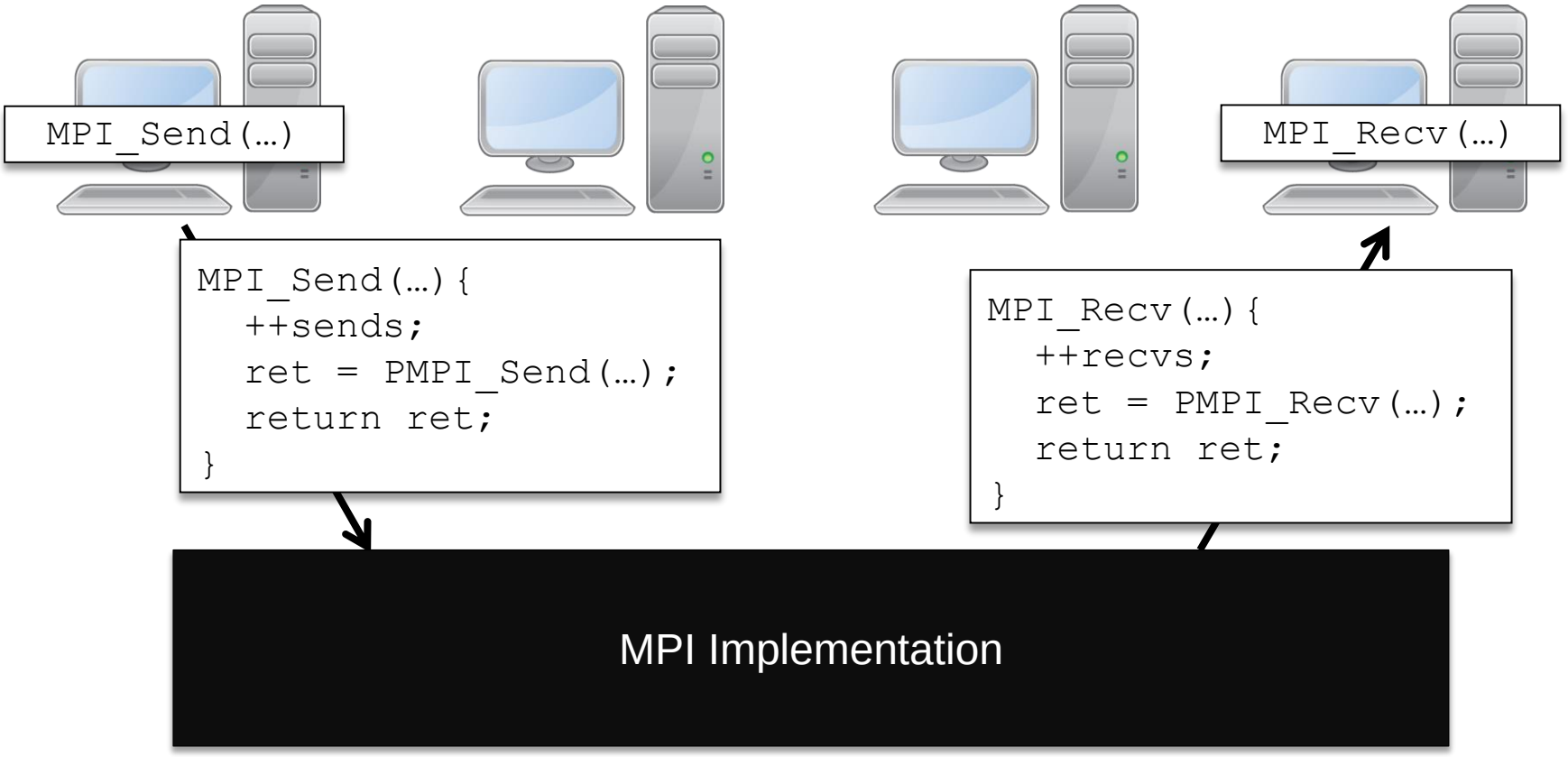
PMPI was very successful

- Performance tools
  - Profilers, tracers, analysis tools, autotuners
- Debugging/correctness tools
- Other tools
  - MPI process replication, power savings, process mapping



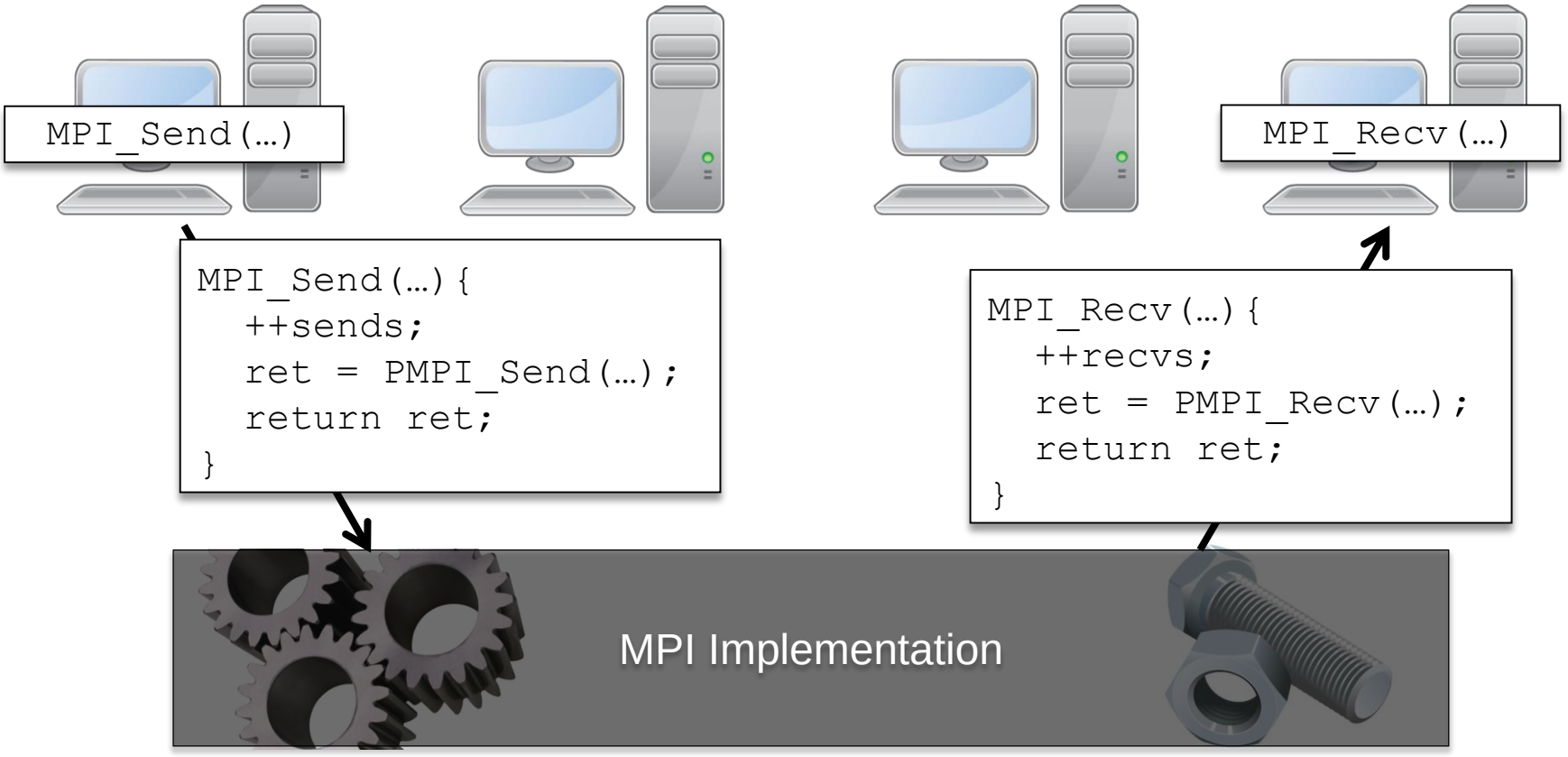


But what happens in the MPI implementation is still a black box…



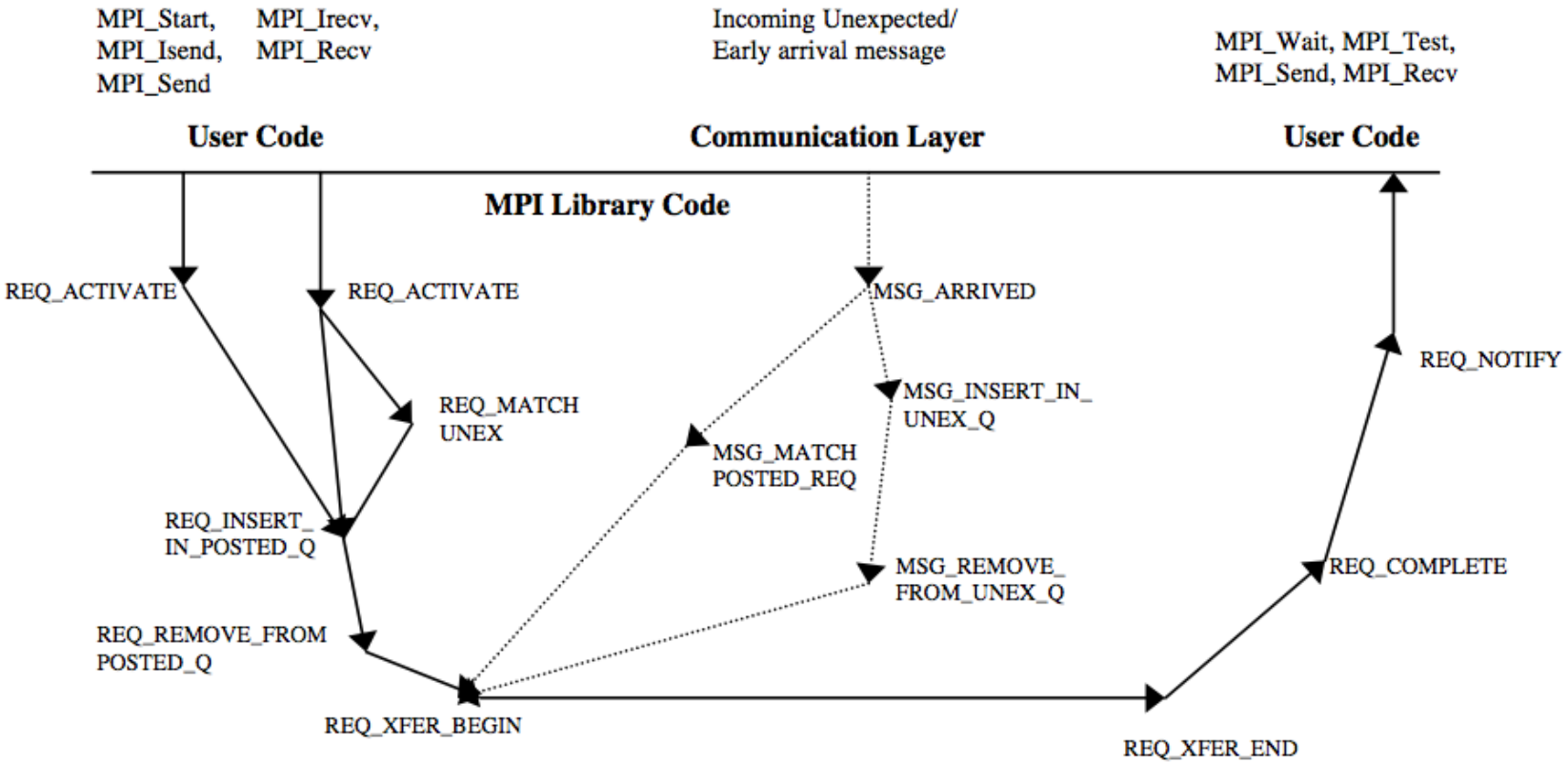
But what happens in the MPI implementation is still a black box…

- Drove the design of the MPI Tools Information Interface (MPI_T)

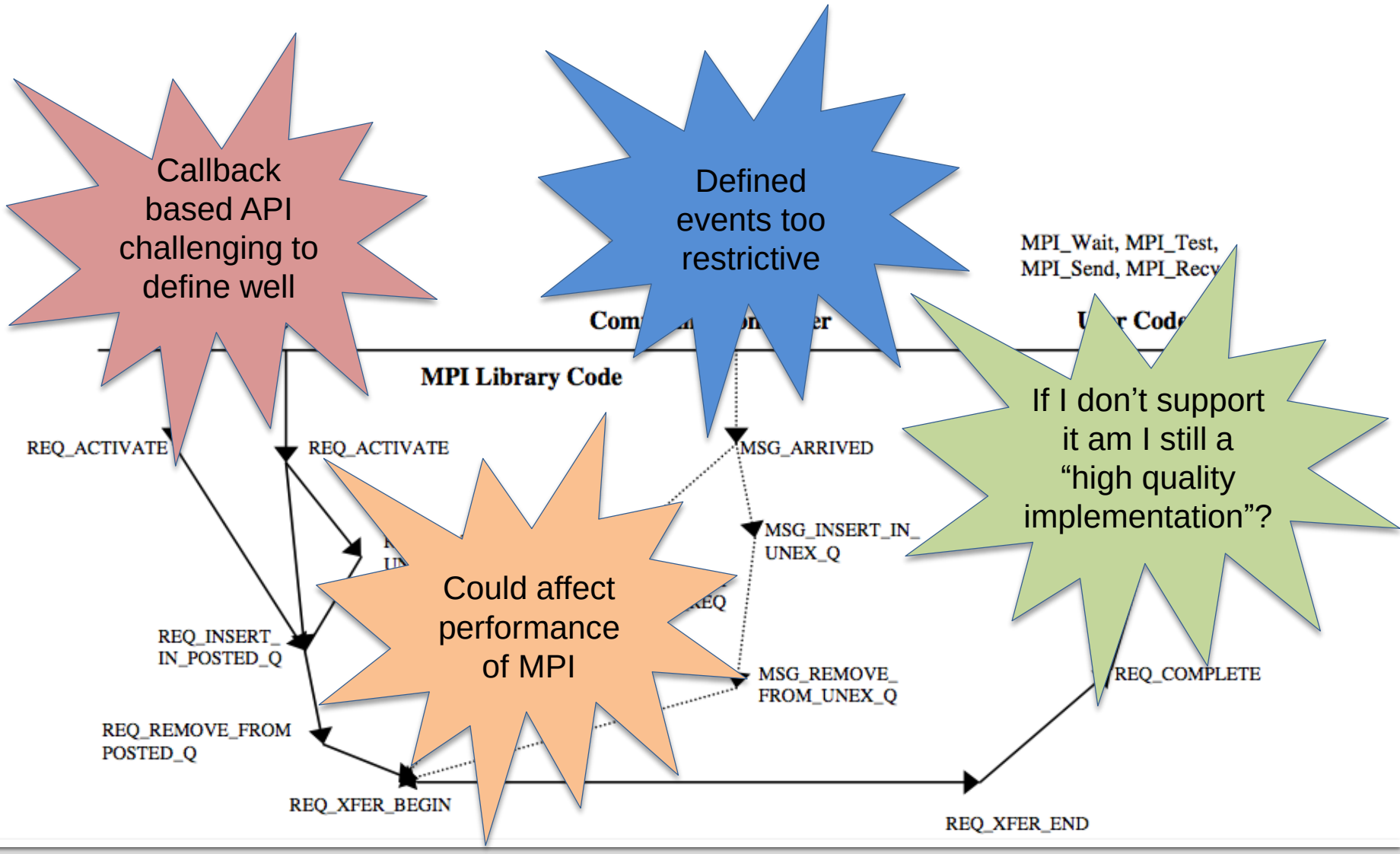


First, a little history…. PERUSE

- PERUSE - MPI performance revealing extensions interface
  - Version 1.0 introduced in 2002
  - Defined events that represent MPI internal information
  - A tool could register for notification of interesting events
  - P2P, collectives, RMA, Spawn, MPI-IO



Why not PERUSE?



Back to the drawing board for MPI_T … Lesson Learned

- The MPI Tools Information Interface (MPI_T for short)
  - Developed over many years, by the folks in the MPI Tools Working Group
  - Led by Martin Schulz

- Included in MPI 3.0 in 2012
  - Now there is a new chapter "Tool Support"
  - Replaces the existing MPI profiling interface chapter
  - PMPI included as a new subchapter (unchanged)

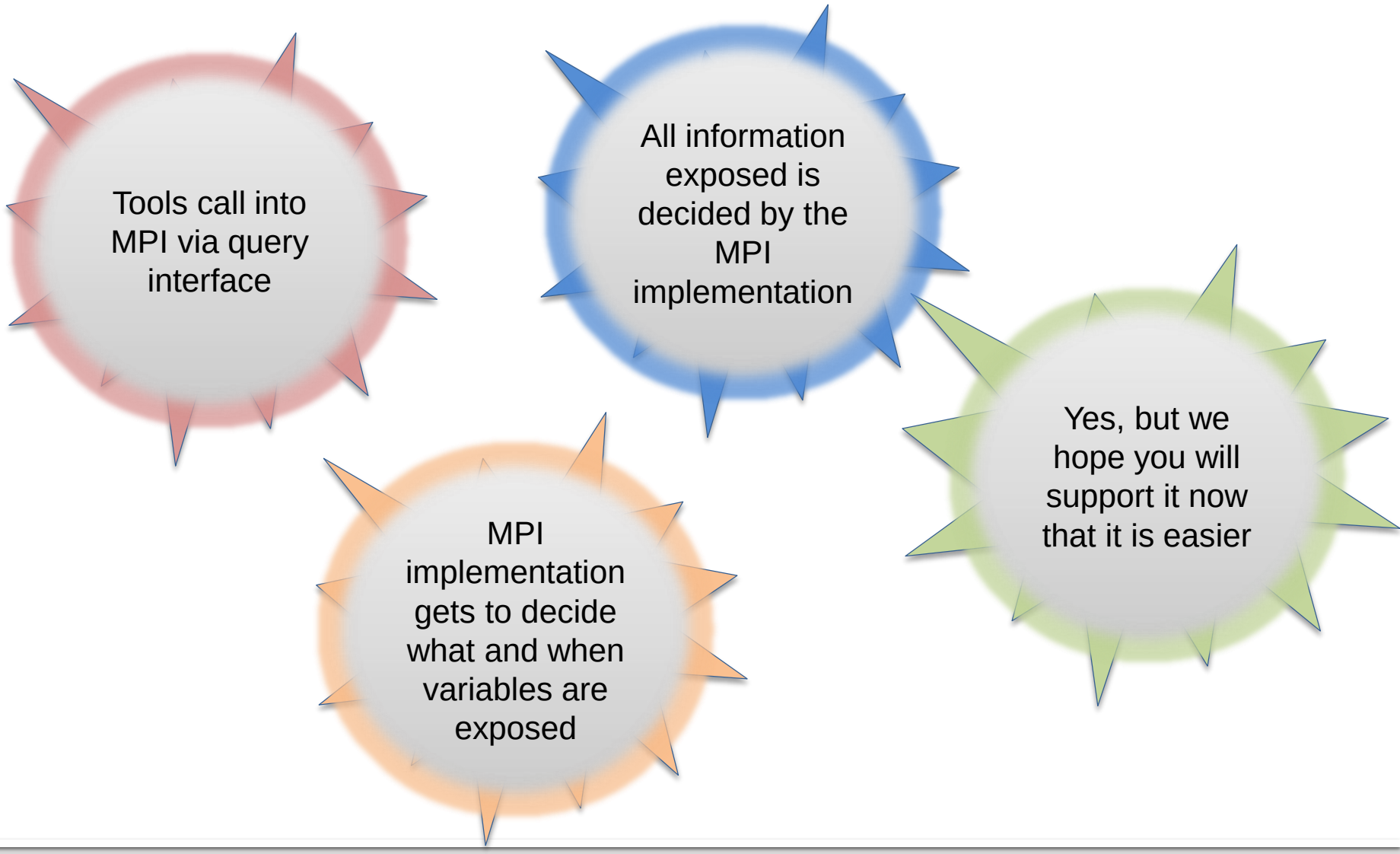
Chapter 14

Tool Support

14.1 Introduction

This chapter discusses interfaces that allow debuggers, performance analyzers, and other tools to extract information about the operation of MPI processes. Specifically, this chapter defines both the MPI profiling interface (Section 14.2), which supports the transparent interception and inspection of MPI calls, and the MPI tool information interface (Section 14.3)

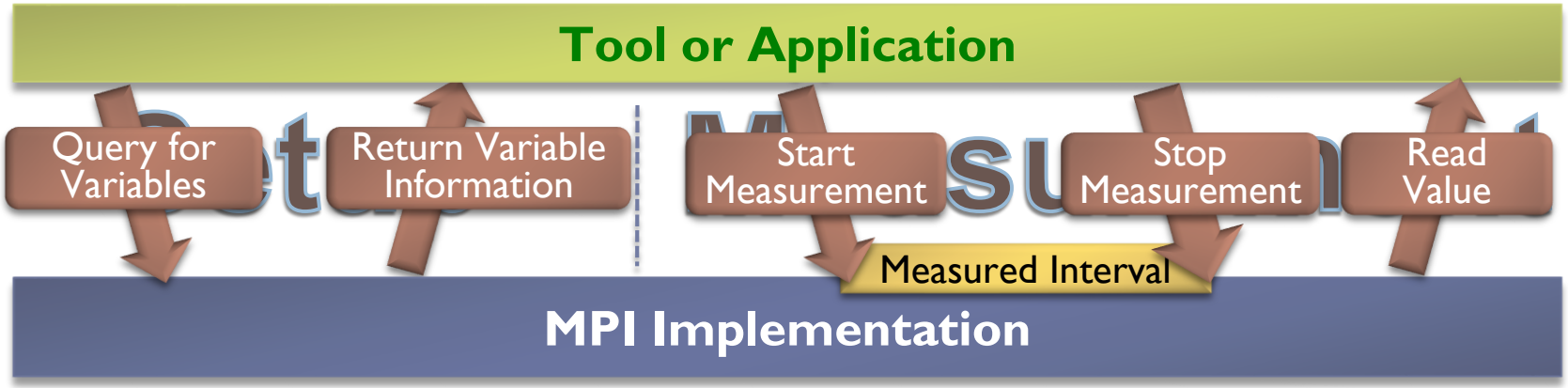
MPI_T is defined as a query interface and the MPI implementation decides what to expose



If nothing is defined, how does MPI_T work?

- Tools must query to see what variables are available before using them
  —EVERY TIME

- MPI implementation decides what variables exist and exposes them through the query interface
  - Variables that exist can be different between …
  - … MPI implementations
  - … compilations of the MPI library (debug vs. production version)
  - … executions of the same application/MPI library
  - … before and after MPI_Init, MPI_Finalize
  - MPI implementations can decide not to provide any variables



Two kinds of variables: performance and control

- Performance Variables
  - Like performance counters
- Examples
  - Number of packets sent
  - Time spent blocking
  - Memory allocated
- Operations
  - Allocate/Free Session
  - Allocate/Free Handle
  - Reset/Write Variable
  - Start Variable
  - Stop Variable
  - Read/Readreset Variable

- Control Variables
  - Configuration
  - Environment variables
- Examples
  - Parameters like Eager Limit
  - Startup control
  - Buffer sizes and management
- Operations
  - Allocate Handle
  - Read/Write Variable
    - Scoping to define to which ranks a configuration change must be applied to
  - Free Handle

We made the interface – will anyone use it?

- This was a bit of a chicken vs the egg problem
  - Tools needed MPI implementations that support it
  - Without tools that use it, not much demand for support in MPI implementations
  - MPI implementation developers said
    - What kinds of variables should we expose?
  - Tool developers said
    - What do you have?
- MPI implementations soon came on board with initial support
  - MPICH, MVAPICH, Open MPI
- And tools followed soon after

2014: First tool on the scene -- VarList

- Simple tool to display all variables offered
  - Extract descriptions and metadata
  - Read default values

OPTIONS	DESCRIPTION
-c	List only Control Variables
-p	List only Performance Variables
-v <VL>	List up to verbosity level=[1,9]
-l	Long list with all information, including descriptions
-m	Do not call `MPI_Init` before listing variables

- Use cases
  - Gather information about which variables are available
  - Documentation of runtime environment

- Tanzima Islam, Kathryn Mohror, and Martin Schulz. Exploring the Capabilities of the New MPI_T Interface. EuroMPI/ASIA '14.

VarList: Control Variables in Open MPI

```
==============================
Control Variables
==============================

Found 1026 control variables
Found 1026 control variables with verbosity <= D/A-9

Variable                                                    VRB   Type   Bind    Scope    Value
-------------------------------------------------------------------------------------------------
...
mpi_ddt_unpack_debug                                        U/A-3 INT    n/a     LOCAL    false
mpi_ddt_pack_debug                                          U/A-3 INT    n/a     LOCAL    false
mpi_ddt_position_debug                                      U/A-3 INT    n/a     LOCAL    false
mpi_ddt_copy_debug                                          U/A-3 INT    n/a     LOCAL    false
dss_buffer_type                                             D/D-8 INT    n/a     ALL      described
dss_buffer_initial_size                                     D/D-8 INT    n/a     ALL      128
dss_buffer_threshold_size                                   D/D-8 INT    n/a     ALL      1024
event                                                       U/D-2 CHAR   n/a     ALL
event_base_verbose                                          D/D-8 INT    n/a     LOCAL    0
event_libevent2021_event_include                            U/A-3 CHAR   n/a     LOCAL    poll
opal_event_include                                          U/A-3 CHAR   n/a     LOCAL    poll
event_libevent2021_major_version                            D/A-9 INT    n/a     UNKNOWN  1
event_libevent2021_minor_version                            D/A-9 INT    n/a     UNKNOWN  9
event_libevent2021_release_version                          D/A-9 INT    n/a     UNKNOWN  0
mpi_param_check                                             D/A-9 INT    n/a     READONLY true
mpi_yield_when_idle                                         D/A-9 INT    n/a     READONLY false
mpi_event_tick_rate                                         D/A-9 INT    n/a     READONLY -1
mpi_show_handle_leaks                                       D/A-9 INT    n/a     READONLY true
mpi_no_free_handles                                         D/A-9 INT    n/a     READONLY false
mpi_show_mpi_alloc_mem_leaks                                D/A-9 INT    n/a     READONLY 0
mpi_show_mca_params                                         D/A-9 CHAR   n/a     READONLY
mpi_show_mca_params_file                                    D/A-9 CHAR   n/a     READONLY
mpi_abort_delay                                             D/A-9 INT    n/a     READONLY 0
mpi_abort_print_stack                                       D/A-9 INT    n/a     READONLY true
...
```

Tanzima Islam, Kathryn Mohror, and Martin Schulz. Exploring the Capabilities of the New MPI_T Interface. EuroMPI/ASIA '14.

VarList: Performance Variables in MVAPICH

```
================================
Performance Variables
================================

Found 25 performance variables
Found 25 performance variables with verbosity <= D/A-9
```

Variable	VRB	Class	Type	Bind	R/O	CNT	ATM
posted_recvq_length	U/D-2	LEVEL	UINT	n/a	YES	YES	NO
unexpected_recvq_length	U/D-2	LEVEL	UINT	n/a	YES	YES	NO
posted_recvq_match_attempts	U/D-2	COUNTER	UNKNOW	n/a	NO	YES	NO
unexpected_recvq_match_attempts	U/D-2	COUNTER	UNKNOW	n/a	NO	YES	NO
time_failed_matching_postedq	U/D-2	TIMER	DOUBLE	n/a	NO	YES	NO
time_matching_unexpectedq	U/D-2	TIMER	DOUBLE	n/a	NO	YES	NO
unexpected_recvq_buffer_size	U/D-2	LEVEL	UNKNOW	n/a	YES	YES	NO
mem_allocated	U/B-1	LEVEL	ULLONG	n/a	YES	YES	NO
mem_allocated	U/B-1	HIGHWAT	ULLONG	n/a	YES	YES	NO
mv2_progress_poll_count	D/B-7	COUNTER	ULONG	n/a	NO	NO	NO
coll_bcast_binomial	U/B-1	COUNTER	ULLONG	n/a	YES	YES	NO
coll_bcast_scatter_doubling_allgather	U/B-1	COUNTER	ULLONG	n/a	YES	YES	NO
coll_bcast_scatter_ring_allgather	U/B-1	COUNTER	ULLONG	n/a	YES	YES	NO
mv2_num_2level_comm_requests	U/D-2	COUNTER	ULONG	n/a	YES	YES	NO
mv2_num_2level_comm_success	U/D-2	COUNTER	ULONG	n/a	YES	YES	NO
mv2_num_shmem_coll_calls	T/B-4	COUNTER	ULONG	n/a	YES	YES	NO
mv2_coll_bcast_binomial	T/B-4	COUNTER	ULLONG	n/a	YES	YES	NO
mv2_coll_bcast_scatter_doubling_allgather	T/B-4	COUNTER	ULLONG	n/a	YES	YES	NO
mv2_coll_bcast_scatter_ring_allgather	T/B-4	COUNTER	ULLONG	n/a	YES	YES	NO
mv2_coll_bcast_scatter_ring_allgather_shm	T/B-4	COUNTER	ULLONG	n/a	YES	YES	NO
mv2_coll_bcast_shmem	T/B-4	COUNTER	ULLONG	n/a	YES	YES	NO
mv2_coll_bcast_knomial_internode	T/B-4	COUNTER	ULLONG	n/a	YES	YES	NO
mv2_coll_bcast_knomial_intranode	T/B-4	COUNTER	ULLONG	n/a	YES	YES	NO
mv2_coll_bcast_mcast_internode	T/B-4	COUNTER	ULLONG	n/a	YES	YES	NO
mv2_coll_bcast_pipelined	T/B-4	COUNTER	ULLONG	n/a	YES	YES	NO

Tanzima Islam, Kathryn Mohror, and Martin Schulz. Exploring the Capabilities of the New MPI_T Interface. EuroMPI/ASIA '14.

2014: We developed Gyan to see what we could learn using MPI_T

- Basic tool to profile MPI_T information
  - Calipers for whole program execution
  - Predefined counters defined through environment variable
    - Identify with Varlist
  - Alternatively: monitor all available variables
- Implemented as a PMPI tool
  - Transparent preloading
  - Data collected and printed at the end of execution
- Following experiments
  - LLNL TLCC cluster (Dual socket Intel Sandybridge nodes and IB)
  - MVAPICH2-2.0a

Tanzima Islam, Kathryn Mohror, and Martin Schulz. Exploring the Capabilities of the New MPI_T Interface. EuroMPI/ASIA '14.

Gyan: Produces simple text output for the execution telling the value of performance variables

Performance profiling for the complete MPI job:

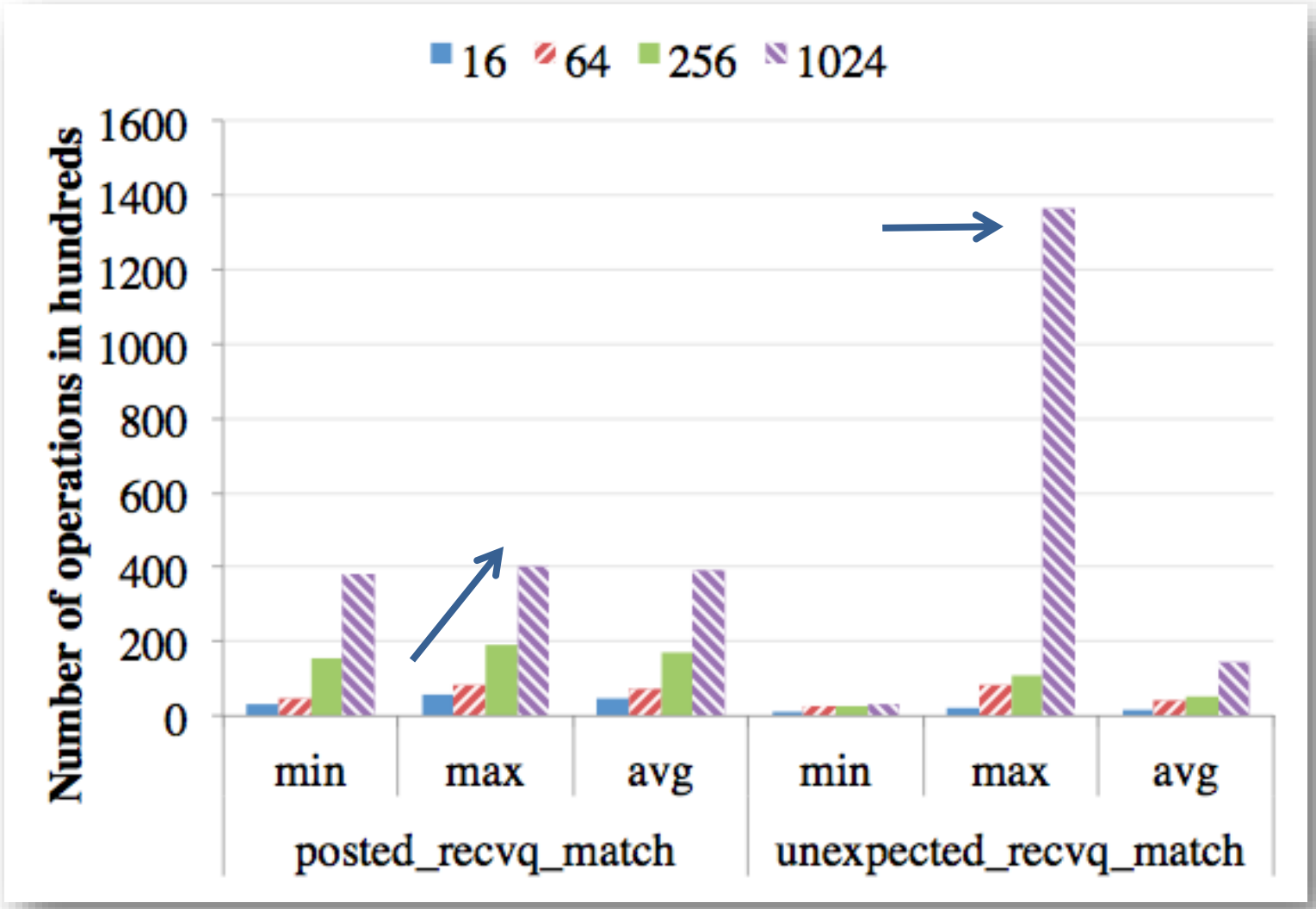
Variable Name	Type	Minimum	Maximum	Average
mem_allocated	LEVEL	2119601	2119601	2119601.00
mem_allocated	HIGHWAT	17488028	17488028	17488028.00
mv2_reg_cache_hits	COUNTER	1205	1205	1205.00
mv2_reg_cache_misses	COUNTER	5	5	5.00
mv2_vbuf_allocated	COUNTER	384	384	384.00
mv2_vbuf_freed	COUNTER	160085	160085	160085.00
mv2_vbuf_available	COUNTER	283	283	283.00
mv2_ud_vbuf_n_allocated	COUNTER	0	0	0.00
mv2_ud_vuf_freed	COUNTER	0	0	0.00
mv2_ud_vbuf_available	COUNTER	0	0	0.00
mv2_progress_poll_count	COUNTER	753207	753207	753207.00
mv2_rdma_ud_retransmit_count	COUNTER	0	0	0.00
coll_bcast_binomial	COUNTER	462	462	462.00
coll_bcast_scatter_doubling_allgather	COUNTER	0	0	0.00
coll_bcast_scatter_ring_allgather	COUNTER	0	0	0.00
mv2_num_2level_comm_requests	COUNTER	1	1	1.00
mv2_num_2level_comm_success	COUNTER	1	1	1.00
mv2_num_shmem_coll_calls	COUNTER	21276	21276	21276.00
mv2_coll_bcast_binomial	COUNTER	0	0	0.00
mv2_coll_bcast_scatter_doubling_allgather	COUNTER	0	0	0.00
mv2_coll_bcast_scatter_ring_allgather	COUNTER	220	220	220.00
mv2_coll_bcast_scatter_ring_allgather_shm	COUNTER	110	110	110.00
mv2_coll_bcast_shmem	COUNTER	3520	3520	3520.00
mv2_coll_bcast_knomial_internode	COUNTER	1760	1760	1760.00
mv2_coll_bcast_knomial_intranode	COUNTER	0	0	0.00
mv2_coll_bcast_mcast_internode	COUNTER	0	0	0.00
mv2_coll_bcast_pipelined	COUNTER	110	110	110.00
mv2_ibv_channel_ctrl_packet_count	COUNTER	0	0	0.00
mv2_ibv_channel_out_of_order_packet_count	COUNTER	0	0	0.00
mv2_ibv_channel_out_of_order_packet_count	COUNTER	0	0	0.00
mv2_rdmafp_ctrl_packet_count	COUNTER	0	0	0.00
mv2_rdmafp_out_of_order_packet_count	COUNTER	0	0	0.00

Tanzima Islam, Kathryn Mohror, and Martin Schulz. Exploring the Capabilities of the New MPI_T Interface. EuroMPI/ASIA '14.

Gyan: Performed case studies using a few potentially interesting variables

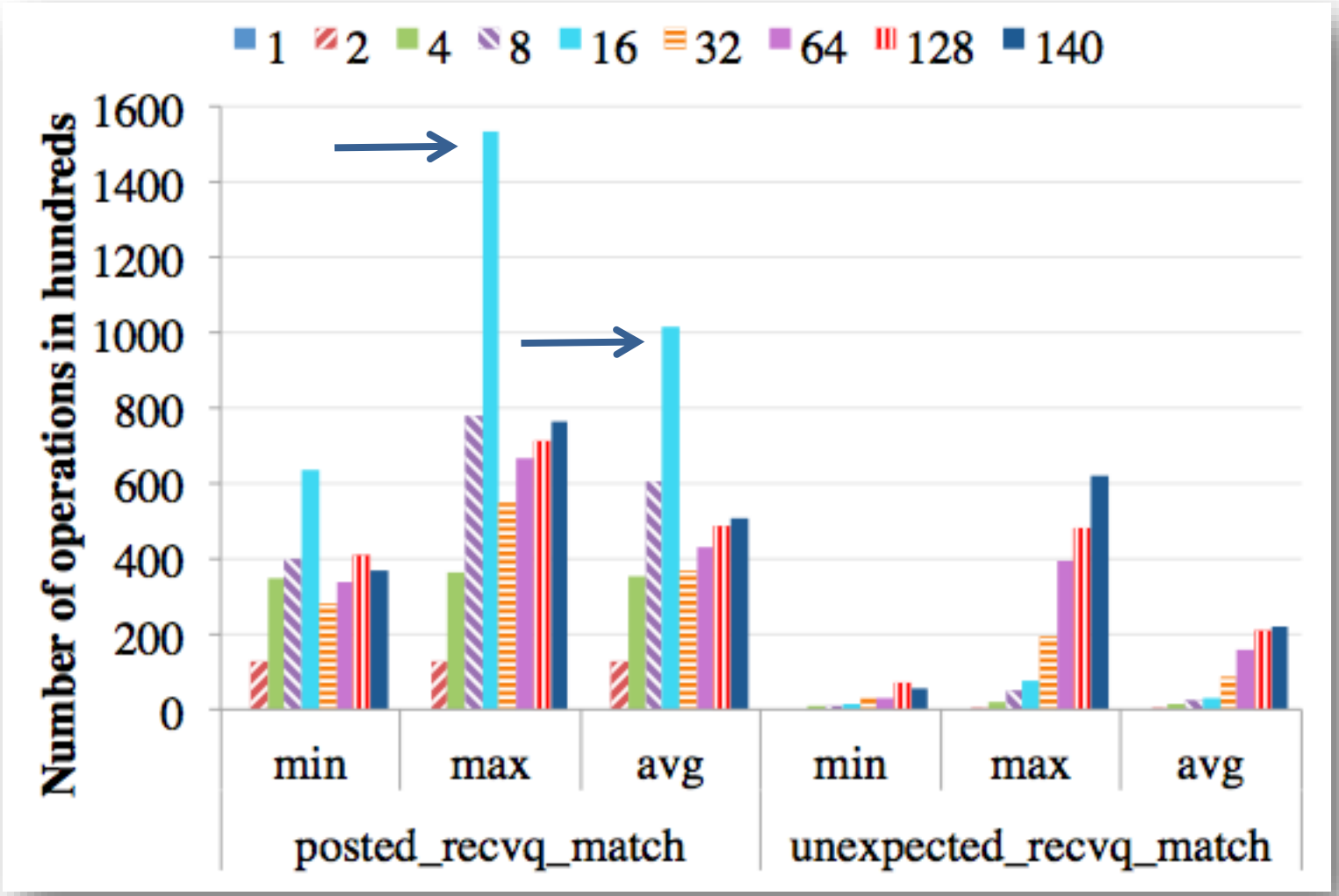
Variable	Description
posted_recvq_match	Counts how many times the queue for receiving expected messages is read.
unexpected_recvq_match	Counts how many times the queue for receiving unexpected messages is read.
mem_allocated_level	Gives the instantaneous memory usage by the library in bytes.
mem_allocated_highwater	Gives the maximum number of bytes ever allocated by the MPI library at a given process for the duration of the application.

Gyan: Receive Queues for NAS BT



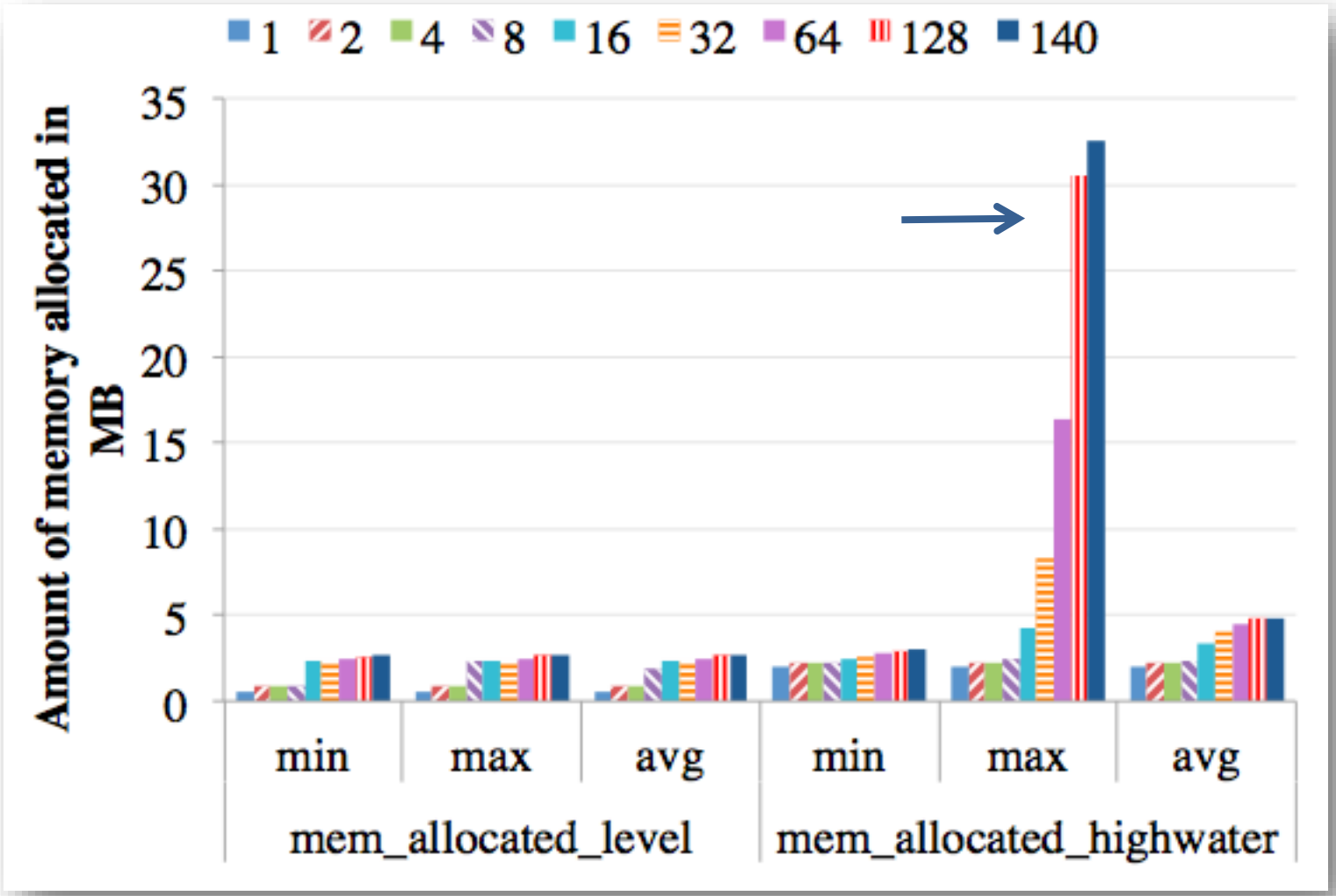
Tanzima Islam, Kathryn Mohror, and Martin Schulz. Exploring the Capabilities of the New MPI_T Interface. EuroMPI/ASIA '14.

Gyan: Receive Queues for NEK5000



Tanzima Islam, Kathryn Mohror, and Martin Schulz. Exploring the Capabilities of the New MPI_T Interface. EuroMPI/ASIA '14.

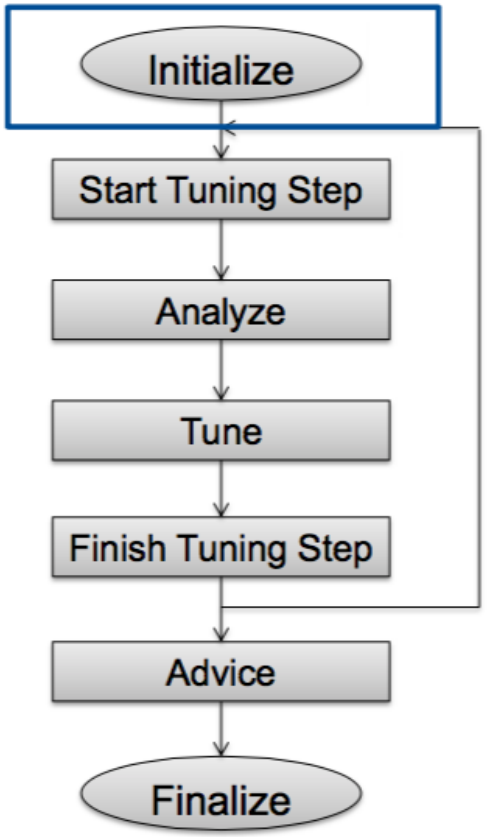
Gyan: Memory Consumption for NEK5000



Tanzima Islam, Kathryn Mohror, and Martin Schulz. Exploring the Capabilities of the New MPI_T Interface. EuroMPI/ASIA '14.

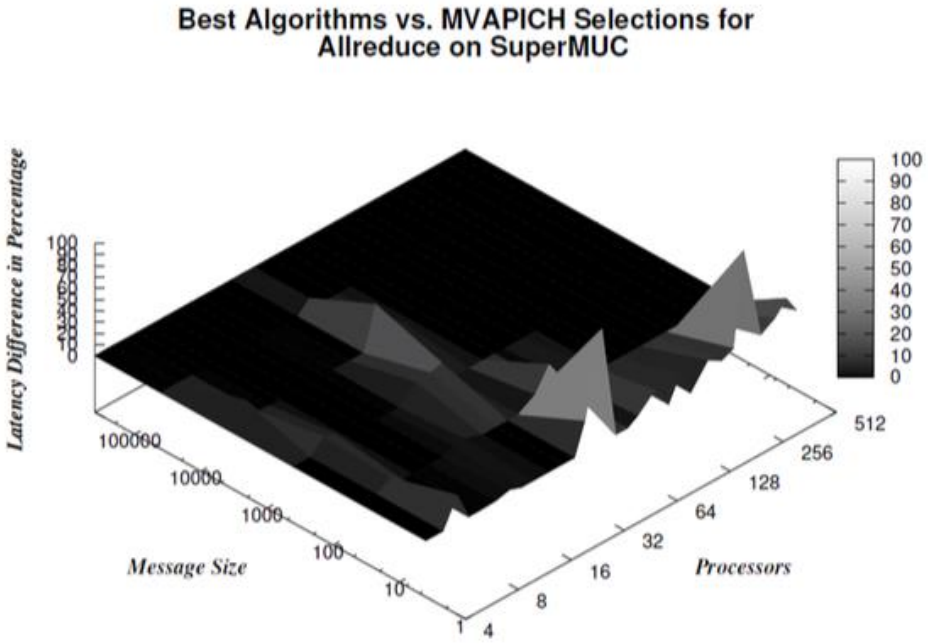
2014: Periscope autotuner used MPI_T to explore parameter space of configuration variables

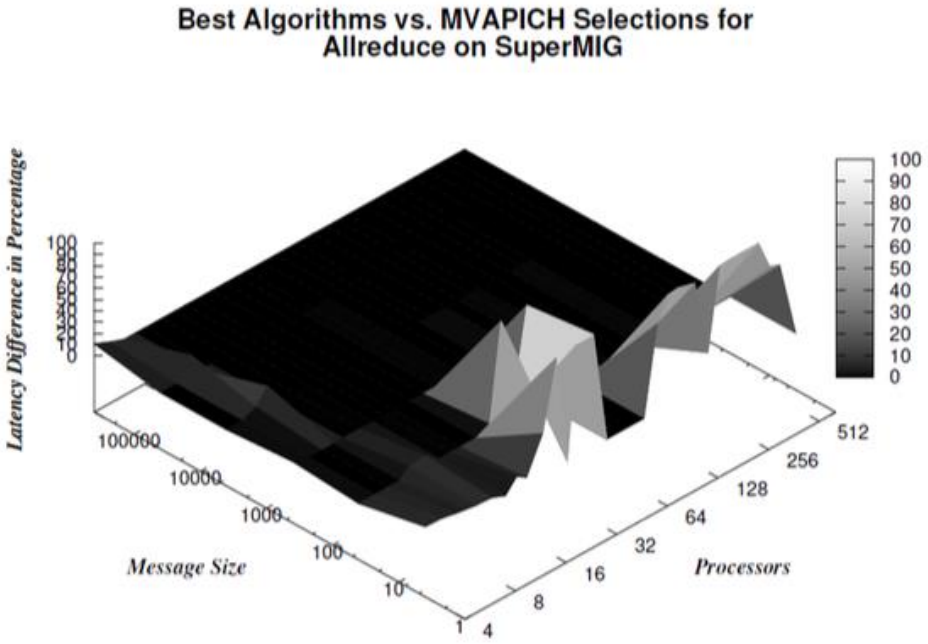
- Explores control variable space to find best values to recommend to user
- Worked with MPICH team to expose new variables
  - And to make some control variables changeable runtime when possible
- Periscope measurement framework has been incorporated into Score-P



- Isaías A. Comprés, "On-line Application-specific Tuning with the Periscope Tuning Framework and the MPI Tools Interface," Petascale Tools Workshop, August 2014.

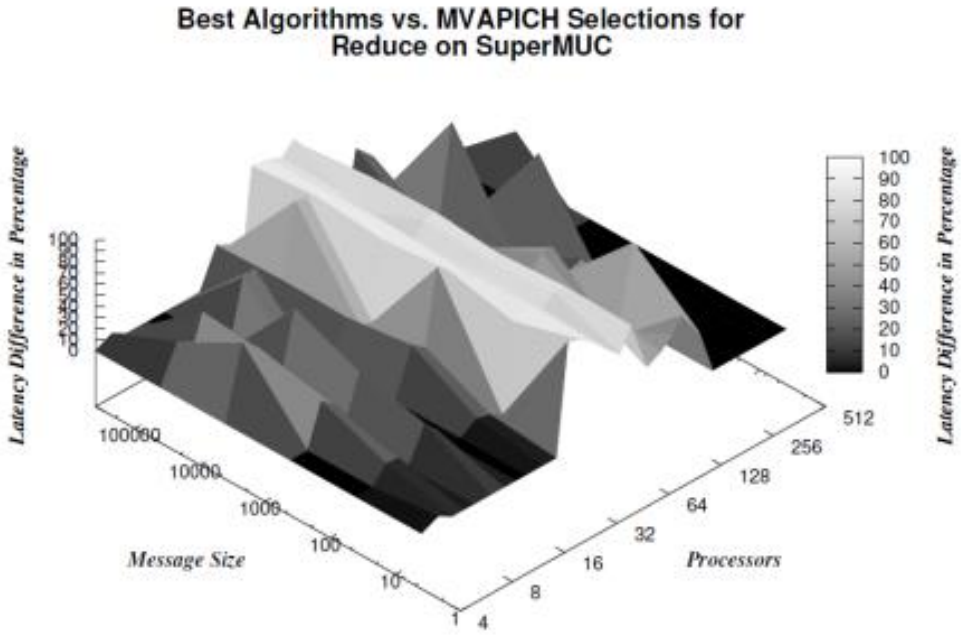
Periscope search finds best algorithm for MPI_Allreduce is not always chosen by default

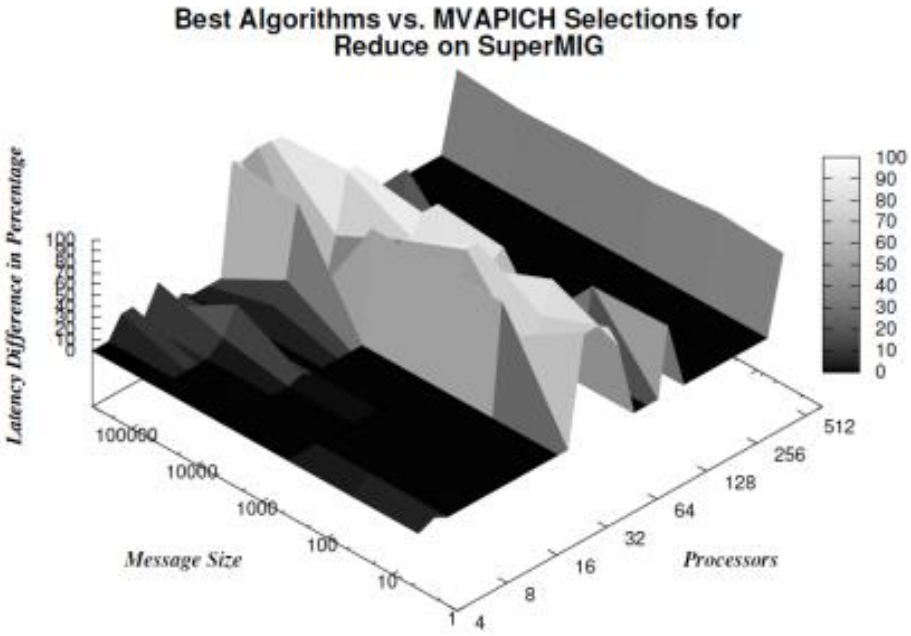




Isaías A. Comprés, “On-line Application-specific Tuning with the Periscope Tuning Framework and the MPI Tools Interface,” Petascale Tools Workshop, August 2014.

Periscope search the default choice for MPI_Reduce showed even more differences in performance over the best

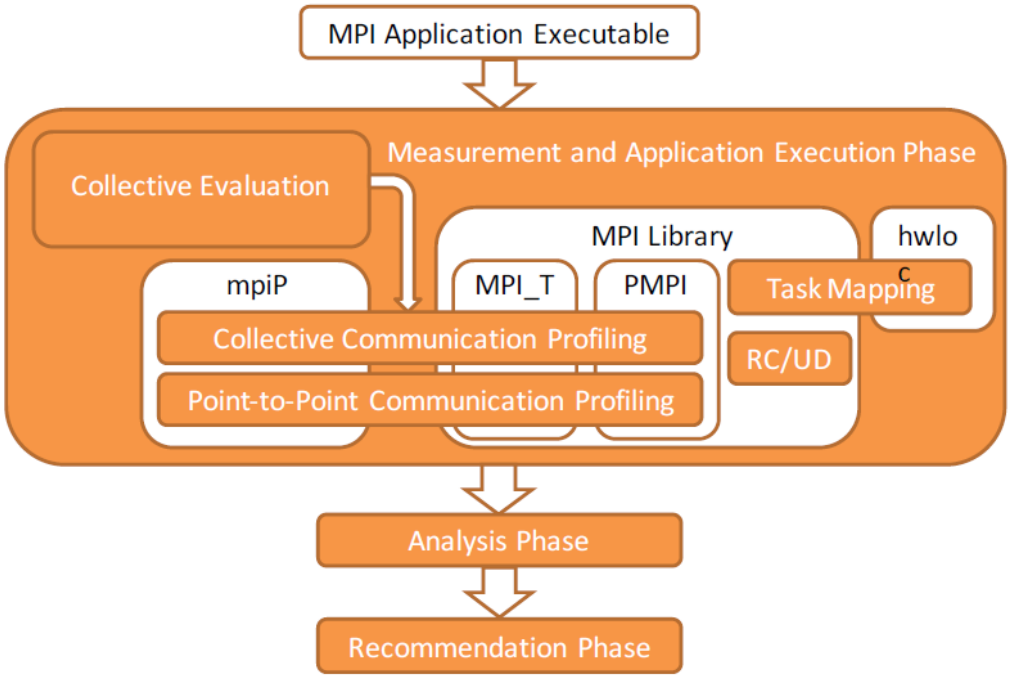




Isaías A. Comprés, “On-line Application-specific Tuning with the Periscope Tuning Framework and the MPI Tools Interface,” Petascale Tools Workshop, August 2014.

2015: MPI Advisor used MPI_T to find optimal settings for performance

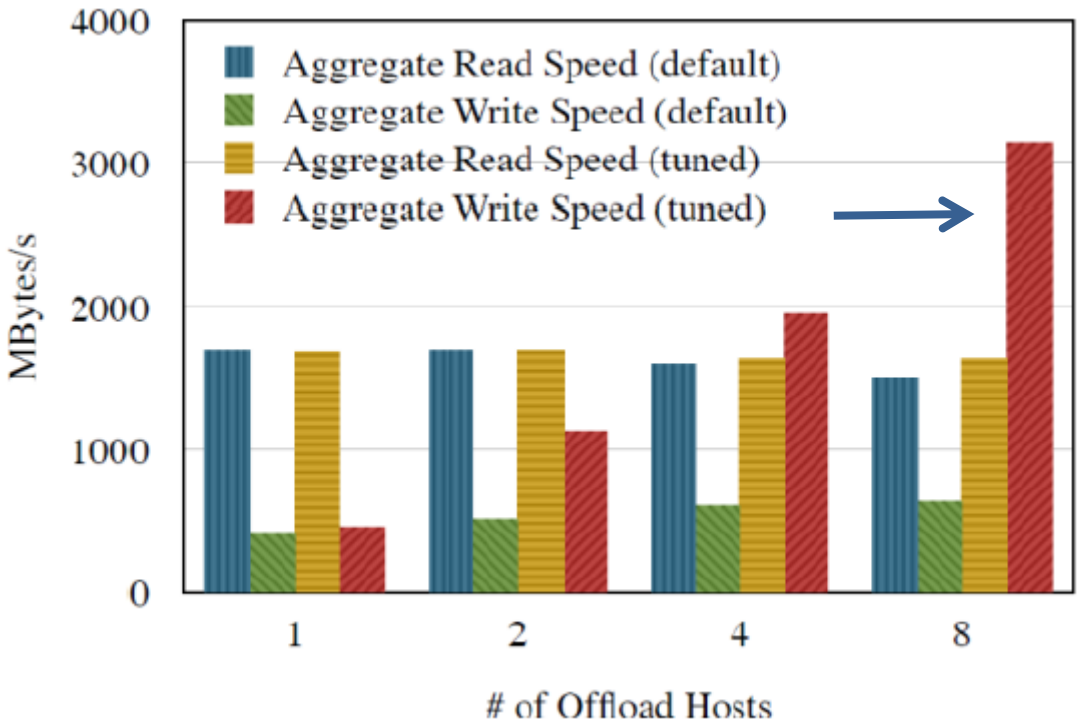
- Approach of MPI Advisor is to recommend optimizations to users



- Used MPI_T control variables to explore settings for
  - Point-to-point protocol threshold (eager limit)
  - Choice of algorithms for collectives
- Working with both MVAPICH2 and Open MPI
  - Continuing work with Open MPI
- Esthela Gallardo, Jerome Vienne, Leonardo Fialho, Patricia Teller, and James Browne. MPI Advisor: a Minimal Overhead Tool for MPI Library Performance Tuning. EuroMPI '15.

MPI Advisor: Eager limit analysis

- Used MPI_T to identify value of eager threshold control variable
- Used mpiP performance data to determine the number and size of messages
- If their algorithm determines that application could benefit from changing eager threshold, they tell user
- CFOUR benchmark
  - Converts disk transactions into distributed memory transactions with MPI
  - MVAPICH2 control variable MV2_IBA_EAGER_THRESHOLD from default 17KB to 256KB
  - ~5x improvement for write



Esthela Gallardo, Jerome Vienne, Leonardo Fialho, Patricia Teller, and James Browne. MPI Advisor: a Minimal Overhead Tool for MPI Library Performance Tuning. EuroMPI '15.

MPI Advisor: Collective algorithms

- For each collective operation there are several algorithms provided by each MPI library that implement the operation
- MPI Advisor determines whether a better algorithm could be used than default and recommends it to the user
- ASP application that uses MPI_Bcast
- Found that by default MVAPICH2 was slower than Intel

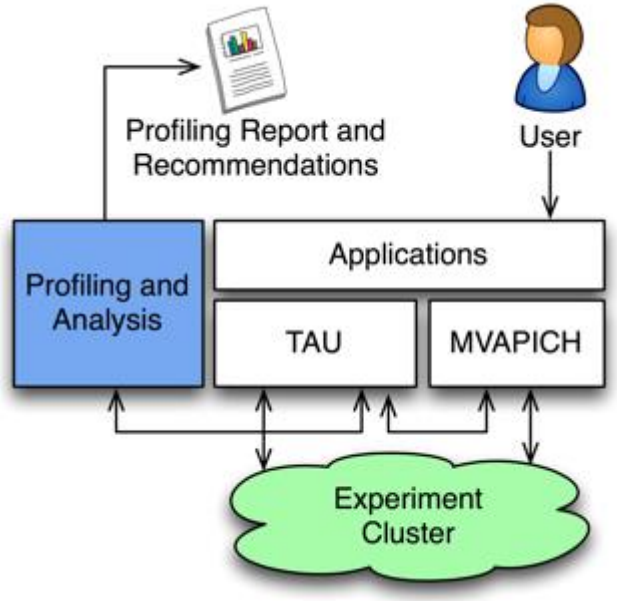
MVAPICH2 Default	MVAPICH2 Tuned	Intel MPI Default
24.45 sec	22.41 sec	22.38 sec

- After changing MV2_INTER_BCAST_TUNING variable performance improved by ~8%, on par with Intel performance

Esthela Gallardo, Jerome Vienne, Leonardo Fialho, Patricia Teller, and James Browne. MPI Advisor: a Minimal Overhead Tool for MPI Library Performance Tuning. EuroMPI '15.

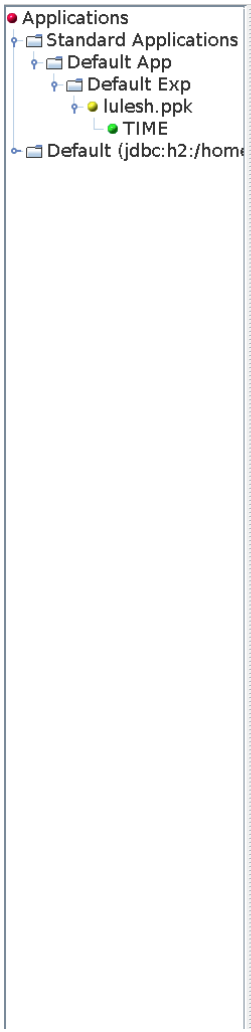
2016: TAU group further advancing the capabilities of tools that use MPI_T

- Tuning and Analysis Utilities (22+ year project)
- Comprehensive performance profiling and tracing
  - Integrated, scalable, flexible, portable
  - Targets all parallel programming/execution paradigms
- http://tau.uoregon.edu
- Recently been collaborating with MVAPICH2 group to use the MPI_T interface to make recommendations for performance to users
  - Co-design effort for implementing new performance and control variables



- Sameer Shende et al., Performance Evaluation using the TAU Performance System, MVAPICH User Group Meeting, August 2016.

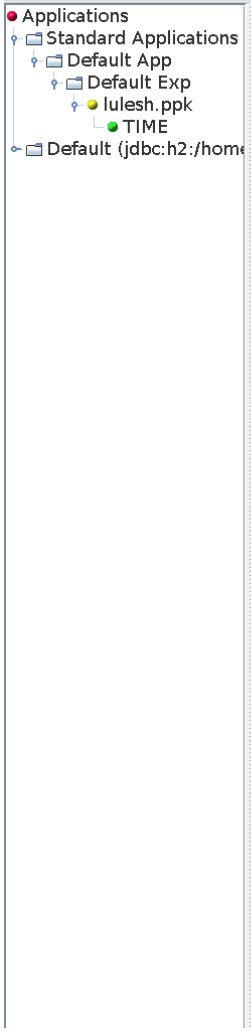
TAU: Collaborating with MVAPICH2 team to expose new performance variables



TrialField	Value
MPI_T PVAR[0]: mem_allocated	Current level of allocated memory within the MPI library
MPI_T PVAR[10]: mv2_num_2level_comm_success	Number of successful 2-level comm creations
MPI_T PVAR[11]: mv2_num_shmem_coll_calls	Number of times MV2 shared-memory collective calls were invoked
MPI_T PVAR[12]: mpit_progress_poll	CH3 RDMA progress engine polling count
MPI_T PVAR[13]: mv2_smp_read_progress_poll	CH3 SMP read progress engine polling count
MPI_T PVAR[14]: mv2_smp_write_progress_poll	CH3 SMP write progress engine polling count
MPI_T PVAR[15]: mv2_smp_read_progress_poll_success	Unsucessful CH3 SMP read progress engine polling count
MPI_T PVAR[16]: mv2_smp_write_progress_poll_succ...	Unsucessful CH3 SMP write progress engine polling count
MPI_T PVAR[17]: rdma_ud_retransmissions	CH3 RDMA UD retransmission count
MPI_T PVAR[18]: mv2_coll_bcast_binomial	Number of times MV2 binomial bcast algorithm was invoked
MPI_T PVAR[19]: mv2_coll_bcast_scatter_doubling_all...	Number of times MV2 scatter+double allgather bcast algorithm was invoked
MPI_T PVAR[1]: mem_allocated	Maximum level of memory ever allocated within the MPI library
MPI_T PVAR[20]: mv2_coll_bcast_scatter_ring_allgather	Number of times MV2 scatter+ring allgather bcast algorithm was invoked
MPI_T PVAR[21]: mv2_coll_bcast_scatter_ring_allgath...	Number of times MV2 scatter+ring allgather shm bcast algorithm was invoked
MPI_T PVAR[22]: mv2_coll_bcast_shmem	Number of times MV2 shmem bcast algorithm was invoked
MPI_T PVAR[23]: mv2_coll_bcast_knomial_internode	Number of times MV2 knomial internode bcast algorithm was invoked
MPI_T PVAR[24]: mv2_coll_bcast_knomial_intranode	Number of times MV2 knomial intranode bcast algorithm was invoked
MPI_T PVAR[25]: mv2_coll_bcast_mcast_internode	Number of times MV2 mcast internode bcast algorithm was invoked
MPI_T PVAR[26]: mv2_coll_bcast_pipelined	Number of times MV2 pipelined bcast algorithm was invoked
MPI_T PVAR[27]: mv2_coll_alltoall_inplace	Number of times MV2 in-place alltoall algorithm was invoked
MPI_T PVAR[28]: mv2_coll_alltoall_bruck	Number of times MV2 brucks alltoall algorithm was invoked
MPI_T PVAR[29]: mv2_coll_alltoall_rd	Number of times MV2 recursive-doubling alltoall algorithm was invoked
MPI_T PVAR[2]: num_malloc_calls	Number of MPIT_malloc calls
MPI_T PVAR[30]: mv2_coll_alltoall_sd	Number of times MV2 scatter-destination alltoall algorithm was invoked
MPI_T PVAR[31]: mv2_coll_alltoall_pw	Number of times MV2 pairwise alltoall algorithm was invoked
MPI_T PVAR[32]: mpit_alltoallv_mv2_pw	Number of times MV2 pairwise alltoallv algorithm was invoked
MPI_T PVAR[33]: mv2_coll_allreduce_shm_rd	Number of times MV2 shm rd allreduce algorithm was invoked
MPI_T PVAR[34]: mv2_coll_allreduce_shm_rs	Number of times MV2 shm rs allreduce algorithm was invoked
MPI_T PVAR[35]: mv2_coll_allreduce_shm_intra	Number of times MV2 shm intra allreduce algorithm was invoked
MPI_T PVAR[36]: mv2_coll_allreduce_intra_p2p	Number of times MV2 intra p2p allreduce algorithm was invoked
MPI_T PVAR[37]: mv2_coll_allreduce_2lvl	Number of times MV2 two-level allreduce algorithm was invoked
MPI_T PVAR[38]: mv2_coll_allreduce_shmem	Number of times MV2 shmem allreduce algorithm was invoked
MPI_T PVAR[39]: mv2_coll_allreduce_mcast	Number of times MV2 multicast-based allreduce algorithm was invoked
MPI_T PVAR[3]: num_calloc_calls	Number of MPIT_calloc calls
MPI_T PVAR[40]: mv2_reg_cache_hits	Number of registration cache hits
MPI_T PVAR[41]: mv2_reg_cache_misses	Number of registration cache misses
MPI_T PVAR[42]: mv2_vbuf_allocated	Number of VBUFs allocated
MPI_T PVAR[43]: mv2_vbuf_allocated_array	Number of VBUFs allocated
MPI_T PVAR[44]: mv2_vbuf_freed	Number of VBUFs freed
MPI_T PVAR[45]: mv2_ud_vbuf_allocated	Number of UD VBUFs allocated
MPI_T PVAR[46]: mv2_ud_vbuf_freed	Number of UD VBUFs freed
MPI_T PVAR[47]: mv2_vbuf_free_attempts	Number of time we attempted to free VBUFs
MPI_T PVAR[48]: mv2_vbuf_free_attempt_success_time	Average time for number of times we sucessfully freed VBUFs
MPI_T PVAR[49]: mv2_vbuf_free_attempt_success_time	Average time for number of times we sucessfully freed VBUFs
MPI_T PVAR[4]: num_memalign_calls	Number of MPIT_memalign calls

Sameer Shende et al., Performance Evaluation using the TAU Performance System, MVAPICH User Group Meeting, August 2016.

TAU: … and new control variables



TrialField	Value
Local Time	2016-08-16T10:11:04-07:00
MPI Processor Name	cerberus.nic.uoregon.edu
MPIR_CVAR_ABORT_ON_LEAKED_HANDLES	If true, MPI will call MPI_Abort at MPI_Finalize if any MPI object handles have been leaked. For example,...
MPIR_CVAR_ALLGATHERV_PIPELINE_MSG_SIZE	The smallest message size that will be used for the pipelined, large-message, ring algorithm in the MPI_...
MPIR_CVAR_ALLGATHER_LONG_MSG_SIZE	For MPI_Allgather and MPI_Allgatherv, the long message algorithm will be used if the send buffer size is ...
MPIR_CVAR_ALLGATHER_SHORT_MSG_SIZE	For MPI_Allgather and MPI_Allgatherv, the short message algorithm will be used if the send buffer size is...
MPIR_CVAR_ALLREDUCE_SHORT_MSG_SIZE	the short message algorithm will be used if the send buffer size is <= this value (in bytes)
MPIR_CVAR_ALLTOALL_MEDIUM_MSG_SIZE	the medium message algorithm will be used if the per-destination message size (sendcount*size(sendtyp...
MPIR_CVAR_ALLTOALL_SHORT_MSG_SIZE	the short message algorithm will be used if the per-destination message size (sendcount*size(sendtype)) ...
MPIR_CVAR_ALLTOALL_THROTTLE	max no. of irecvs/isends posted at a time in some alltoall algorithms. Setting it to 0 causes all irecvs/isen...
MPIR_CVAR_ASYNC_PROGRESS	If set to true, MPICH will initiate an additional thread to make asynchronous progress on all communicati...
MPIR_CVAR_BCAST_LONG_MSG_SIZE	Let's define short messages as messages with size < MPIR_CVAR_BCAST_SHORT_MSG_SIZE, and mediu...
MPIR_CVAR_BCAST_MIN_PROCS	Let's define short messages as messages with size < MPIR_CVAR_BCAST_SHORT_MSG_SIZE, and mediu...
MPIR_CVAR_BCAST_SHORT_MSG_SIZE	Let's define short messages as messages with size < MPIR_CVAR_BCAST_SHORT_MSG_SIZE, and mediu...
MPIR_CVAR_CH3_EAGER_MAX_MSG_SIZE	This cvar controls the message size at which CH3 switches from eager to rendezvous mode.
MPIR_CVAR_CH3_ENABLE_HCOLL	If true, enable HCOLL collectives.
MPIR_CVAR_CH3_INTERFACE_HOSTNAME	If non-NULL, this cvar specifies the IP address that other processes should use when connecting to this pr...
MPIR_CVAR_CH3_NOLOCAL	If true, force all processes to operate as though all processes are located on another node. For example,...
MPIR_CVAR_CH3_ODD_EVEN_CLIQUES	If true, odd procs on a node are seen as local to each other, and even procs on a node are seen as local t...
MPIR_CVAR_CH3_PORT_RANGE	The MPIR_CVAR_CH3_PORT_RANGE environment variable allows you to specify the range of TCP ports ...
MPIR_CVAR_CH3_RMA_ACC_IMMED	Use the immediate accumulate optimization
MPIR_CVAR_CH3_RMA_GC_NUM_COMPLETED	Threshold for the number of completed requests the runtime finds before it stops trying to find more co...
MPIR_CVAR_CH3_RMA_GC_NUM_TESTED	Threshold for the number of RMA requests the runtime tests before it stops trying to check more reques...
MPIR_CVAR_CH3_RMA_LOCK_IMMED	Issue a request for the passive target RMA lock immediately. Default behavior is to defer the lock reque...
MPIR_CVAR_CH3_RMA_MERGE_LOCK_OP_UNLOCK	Enable/disable an optimization that merges lock, op, and unlock messages, for single-operation passive ta...
MPIR_CVAR_CH3_RMA_NREQUEST_NEW_THRESHOLD	Threshold for the number of new requests since the last attempt to complete pending requests. Higher ...
MPIR_CVAR_CH3_RMA_NREQUEST_THRESHOLD	Threshold at which the RMA implementation attempts to complete requests while completing RMA oper...
MPIR_CVAR_CHOP_ERROR_STACK	If >0, truncate error stack output lines this many characters wide. If 0, do not truncate, and if <0 use a ...
MPIR_CVAR_COLL_ALIAS_CHECK	Enable checking of aliasing in collective operations
MPIR_CVAR_COMM_SPLIT_USE_QSORT	Use qsort(3) in the implementation of MPI_Comm_split instead of bubble sort.
MPIR_CVAR_CTXID_EAGER_SIZE	The MPIR_CVAR_CTXID_EAGER_SIZE environment variable allows you to specify how many words in th...
MPIR_CVAR_DEBUG_HOLD	If true, causes processes to wait in MPI_Init and MPI_Initthread for a debugger to be attached. Once the ...
MPIR_CVAR_DEFAULT_THREAD_LEVEL	Sets the default thread level to use when using MPI_INIT.
MPIR_CVAR_DUMP_PROVIDERS	If true, dump provider information at init
MPIR_CVAR_ENABLE_COLL_FT_RET	DEPRECATED! Will be removed in MPICH-3.2 Collectives called on a communicator with a failed process...
MPIR_CVAR_ENABLE_SMP_ALLREDUCE	Enable SMP aware allreduce.
MPIR_CVAR_ENABLE_SMP_BARRIER	Enable SMP aware barrier.
MPIR_CVAR_ENABLE_SMP_BCAST	Enable SMP aware broadcast (See also: MPIR_CVAR_MAX_SMP_BCAST_MSG_SIZE)
MPIR_CVAR_ENABLE_SMP_COLLECTIVES	Enable SMP aware collective communication.
MPIR_CVAR_ENABLE_SMP_REDUCE	Enable SMP aware reduce.
MPIR_CVAR_ERROR_CHECKING	If true, perform checks for errors, typically to verify valid inputs to MPI routines. Only effective when M...
MPIR_CVAR_GATHERV_INTER_SSEND_MIN_PROCS	Use Ssend (synchronous send) for intercommunicator MPI_Gatherv if the "group B" size is >= this value....
MPIR_CVAR_GATHER_INTER_SHORT_MSG_SIZE	use the short message algorithm for intercommunicator MPI_Gather if the send buffer size is < this value...
MPIR_CVAR_GATHER_VSMALL_MSG_SIZE	use a temporary buffer for intracommunicator MPI_Gather if the send buffer size is < this value (in bytes...
MPIR_CVAR_IBA_EAGER_THRESHOLD	0 (old) -> 204800 (new), This set the switch point between eager and rendezvous protocol

Sameer Shende et al., Performance Evaluation using the TAU Performance System, MVAPICH User Group Meeting, August 2016.

TAU: Total memory used for VBUFs improved by setting control variable, signficantly reduces MPI memory footprint

TAU: ParaProf: Context Events for: node 0 - mpit_withoutcvar_bt.C.1k.ppk

Name △	MaxValue	MinValue	MeanValue	Std. Dev.	NumSamples	Total
mv2_total_vbuf_memory (Total amount of memory in bytes used for VBUFs)	3,313,056	3,313,056	3,313,056	0	1	3,313,056

TAU: ParaProf: Context Events for: node 0 - bt-mz.E.vbuf_pool_16.1k.ppk

Name △	MaxValue	MinValue	MeanValue	Std. Dev.	NumSamp...	Total
mv2_total_vbuf_memory (Total amount of memory in bytes used for VBUFs)	1,815,056	1,815,056	1,815,056	0	1	1,815,056
mv2_ud_vbuf_allocated (Number of UD VBUFs allocated)	0	0	0	0	0	0
mv2_ud_vbuf_available (Number of UD VBUFs available)	0	0	0	0	0	0
mv2_ud_vbuf_freed (Number of UD VBUFs freed)	0	0	0	0	0	0
mv2_ud_vbuf_inuse (Number of UD VBUFs inuse)	0	0	0	0	0	0
mv2_ud_vbuf_max_use (Maximum number of UD VBUFs used)	0	0	0	0	0	0
mv2_vbuf_allocated (Number of VBUFs allocated)	160	160	160	0	1	160
mv2_vbuf_available (Number of VBUFs available)	94	94	94	0	1	94
mv2_vbuf_freed (Number of VBUFs freed)	5,479	5,479	5,479	0	1	5,479
mv2_vbuf_inuse (Number of VBUFs inuse)	66	66	66	0	1	66
mv2_vbuf_max_use (Maximum number of VBUFs used)	66	66	66	0	1	66
num_calloc_calls (Number of MPIT_calloc calls)	89	89	89	0	1	89
num_free_calls (Number of MPIT_free calls)	130	130	130	0	1	130
num_malloc_calls (Number of MPIT_malloc calls)	1,625	1,625	1,625	0	1	1,625
num_memalign_calls (Number of MPIT_memalign calls)	56	56	56	0	1	56
num_memalign_free_calls (Number of MPIT_memalign_free calls)	0	0	0	0	0	0

TAU: ParaProf Manager

- Applications
  - Standard Applications
    - Default App
      - Default Exp
        - bt-mz.E.vbuf_pool_16.1k.pp
          - TIME

TrialField	Value
MPI Processor Name	c526-502.stampede.tacc.utexas.edu
MPIR_CVAR_VBUF_POOL_SIZE	0 (old) -> 16 (new), This set the size of the VBUF pool

Sameer Shende et al., Performance Evaluation using the TAU Performance System, MVAPICH User Group Meeting, August 2016.

MPI_T: What's the verdict?

- Kind of a chicken and egg problem
- Little by little folks are starting to develop tools and expose more variables
  - Collaborations like those between TAU and MVAPICH2, Periscope and MPICH, MPI Advisor and Open MPI
  - More tools in the future

- Not sure how the undefined variables will play out
  - Will tool developers create a unified taxonomy of variables (like PAPI does for hardware counters)?

What's next for the Tools Working Group for performance tools?

- Extend MPI_T to support event type variables
  - Call backs to notify tool of event occurrence
  - E.g., when message placed in queue for non-blocking communication
  - Again not defined what the events will be

- Fix the original profiling interface PMPI
  - Only one tool can use it at a time
  - Prohibits layering of tools and libraries
  - Complete redesign of the interface

- https://github.com/mpiwg-tools/tools-issues/wiki


Lawrence Livermore
National Laboratory