

Intel MPI Tools

A brief Overview

Christian Terboven <terboven@rz.rwth-aachen.de>

20.03.2012 / Aachen, Germany

Stand: 19.03.2012

Version 2.3

▶ **Use it:** `module switch openmpi intelmpi`

- ▶ Current version: 4.0
- ▶ Also available on Windows

▶ **Intel Processor Information Utility:** `cpuinfo`

▶ **Debug Aid:** environment variable `I_MPI_DEBUG=<level>`

<level>	Description
0	No debugging information, default.
1	Error Diagnostics.
2	Confirm which <code>I_MPI_DEVICE</code> was used.
3	Output effective MPI rank, pid and node mapping.
4	Print process pinning information.
5	Print Intel MPI-specific environment variables.
> 5	Add extra levels of verbosity, up to more than you can take...

▶ **Performance Tuning Aid: `I_MPI_STATS=[n-]m`**

n, m	Description
1	Output the amount of data sent by each process.
2	Output the number of calls and amount of transferred data.
3	Output statistics combined according to the actual arguments.
4	Output statistics defined by a buckets list.
10	Output collective operation statistics for all communication contexts.

- ▶ `I_MPI_STATS_FILE=<filename>`: define the statistics output file.

▶ Performance Tuning Aid: Intel MPI Tracing facilities

- ▶ `module load intelitac`
- ▶ Set `VT_ROOT` env. var to `/opt/intel/itac/8.0.3.007`
- ▶ Compile with `-t` or `-trace` using the Intel C/C++/Fortran compilers.
- ▶ Execute with `mpiexec -trace programname`
- ▶ A file named `programname.stf` will be written
- ▶ Open the file with `traceanalyzer programname.stf`

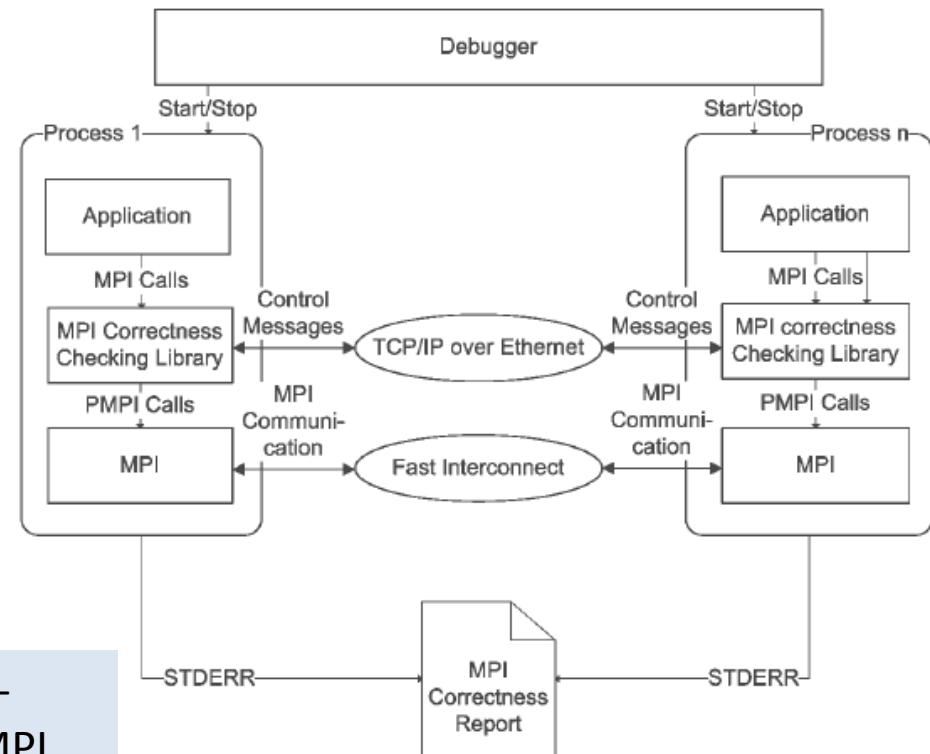
▶ Analysis capabilities similar to Vampir, but the tracing facilities of Intel MPI are not as scalable as those from vampirtrace!

▶ Correctness Checking Aid: Intel MPI Checker

- ▶ `module load intelitac`
- ▶ Set `VT_ROOT` env. var to `/opt/intel/itac/8.0.3.007`
- ▶ Compile with `-check_mpi` using the Intel C/C++/Fortran compilers.
- ▶ Execute with

```
mpiexec -check-mpi a.out
```

- ▶ Use the smallest, possible
- ▶ configuration, leading to a
- ▶ short runtime
- ▶ Be aware of
- ▶ DEADLOCK-TIMEOUT: 60s



P. Ohly and W. Krotz-Vogel: Automated MPI Correctness Checking

Fig. 1. Overview of TAC MPI Correctness Checking

Thank you for your attention.